

DOS III RELOCATING LOADER
(24308A - REV. 1432)

BINARY TAPE	24308-60001
SOURCE TAPES	24308-80001
	24308-80002
	24308-80003
	24308-80004
	24308-80005
	24308-80006
	24308-80007
SOURCE LISTING	24308-90001

PAGE 0001

0001

ASMB,L,X,C,N,B,B

** NO ERRORS*

```

0001          ASMB,L,X,C,N,R,B
0003 00000      NAM LOADR,3,0
0004          EXT EXEC
0005          ENT LOADR
0006          SUP

```

0007*

0008*

0009*

0010*

0011*

0012*

0013*

0014*

0015*

0016*

0017*

0018*

0019*

0020*

0021*

0022*

0023*

0024*

0025*

0026*

0027*

0028*

0029*

0030*

0031*

0032*

THE DISC OPERATING SYSTEM RELOCATING LOADER IS USED TO RELOCATE BINARY PROGRAMS TO ABSOLUTE CODE THAT ARE TO BE RUN IN THE USER AREA OF THE SYSTEM. THE RELOCATABLE BINARY MAY COME FROM THE D.O.S. ASSEMBLER, FORTRAN, AND/OR ALGOL. THESE BINARY PROGRAMS MAY BE LOADED FROM BINARY TAPES, JOB BINARY FILE OR USER SPECIFIED FILE(S).

PROGRAMS CAN BE MADE TO BE MADE UP OF A -MAIN- PROGRAM WITH OR WITHOUT PROGRAM SEGMENTS. PROGRAMS CAN BE MADE TO RUN UNDER CONTROL OF -DEBUG- ROUTINE. REQUESTED FORTRAN RELOCATABLE BINARY LIBRARY ROUTINES ARE AUTOMATICALLY APPENDED TO THE END OF THE PROGRAMS. A FRESH COPY OF -DEBUG- SHALL BE APPENDED TO EACH SEGMENT BY THE LOADER IF THE DEBUG OPTION IS SELECTED. TO INSURE THAT THE SEGMENT, AS WELL AS -MAIN- , RUNS UNDER CONTROL OF -DEBUG-

THE ABSOLUTE CODE IS PLACED ON THE DISC IMMEDIATELY FOLLOWING THE END OF THE LAST USER FILE. THE ABSOLUTE CODE CANNOT EXCEED THE USER AREA CORE SIZE. THE BASE PAGE LINKAGES, IF ANY ARE REQUIRED, ARE PLACED ON THOSE SECTORS OF DISC TRACK WHICH FOLLOW THE -MAIN- OR -SEGMENT- FOR THE PROGRAM.

THE USER DIRECTORY IS UPDATED FOR EACH PROGRAM -MAIN- OR -SEGMENT- AND THE USER AREA BOUNDARY VALUE IS UPDATED.

PAGE 0003 #01 DISC OPERATING SYSTEM RELOCATING LOADER

0034* THE RELOCATING LOADER PERMITS SEVERAL LOADING OPTIONS ENTERED
0035* VIA,
0036* :PROG,LOADR(,L,M,N,O,P)
0037*
0038* WHERE
0039*
0040* L IS SECONDARY INPUT DEVICE LOGICAL UNIT. IF L
0041* NOT SPECIFIED THEN INPUT FROM JBIN AREA ONLY.
0042* IF 2, THEN FROM JBIN AND USER FILES ONLY
0043* IF NOT 0 OR 2, THEN FROM JBIN, USER FILES, AND
0044* PAPER TAPE FROM LOGICAL UNIT L
0045* M IS LIST OUTPUT DEVICE LOGICAL UNIT NUMBER. IF NOT
0046* SPECIFIED, THEN 6 IS ASSUMED.
0047* N IS NO DEBUG/DEBUG OPTION. 0 IS ASSUMMED IF NOT
0048* SPECIFIED.
0049* O IS CURRENT/BASE PAGE
0050* LINKING OPTION. 0 IS
0051* ASSUMED IF NOT SPECIFIED.
0052* P IS PROGRAM BOUNDS OPTION. 0 IS
0053* ASSUMED IF NOT SPECIFIED.

0055* THE RELOCATABLE LOADER FUNCTIONS AS FOLLOWS:

- 0056*
- 0057* (1). PROCESSING OF CONTROL INPUT SPECIFIED FILES (INCLUDES
- 0058* JOB BINARY FILE AND SPECIFIED USER FILES)
- 0059* (2). PROCESSING BINARY PROGRAMS INPUT FROM THE SECONDARY
- 0060* INPUT DEVICE.
- 0061* (3). SEARCH RELOCATABLE LIBRARY TO ATTEMPT TO SATISFY
- 0062* UNDEFINED EXTERNAL REFERENCES.
- 0063* (4). GENERATE ABSOLUTE CODE ON DISC FOR MAIN AND SEGMENTS
- 0064* AND UPDATE USER DIRECTORY AND USER-WORK AREA BOUNDARY.
- 0065*

0066* TWO TABLES ARE GENERATED BY THE RELOCATING LOADER WHEN

0067* ENCOUNTERING NAM AND ENT/EXT RECORDS OF THE RELOCATABLE BINARY

0068* PROGRAMS ON USER FILE, JOB BINARY FILES, AND PROGRAMS INPUT VIA

0069* SECONDARY INPUT DEVICE, THE TABLES ARE :

- 0070*
- 0071* (1). IDENT TABLE FOR PROGRAM NAME NAME ENTRIES
- 0072* (2). LOADER SYMBOL TABLE FOR ENT/EXT ENTRIES
- 0073*

0074* IDENT TABLE FORMAT(ID)

- 0075*
- 0076* WORD 1: ID1 - NAME 1,2
- 0077* WORD 2: ID2 - NAME 3,4
- 0078* WORD 3: ID3 - NAME 5,USAGE FLAG
- 0079* WORD4: ID4 - BIT15 IS DISC LOC FLAG; REM FOR COMMON LENGTH
- 0080* WORD 5: ID5 - DISC ADDRESS
- 0081* WORD 6: ID6 - (15)=MAIN/SUBROUTINE ; (6=0)=TYPE
- 0082* WORD 7: ID7 - DISC LENGTH ; USER MAIN IDENT ADDR FOR SEGMENTS
- 0083*

0084* LOADER SYMBOL TABLE FORMAT(LST)

- 0085*
- 0086* WORD 1: LST1 - NAME 1,2
- 0087* WORD 2: LST2 - NAME 3,4
- 0088* WORD 3: LST3 - NAME 5,ORDINAL
- 0089* WORD 4: LST4 - IDENT ADDRESS
- 0090* WORD 5: LST5 - BP LINK ADDRESS
- 0091*

0092* PROGRAM TYPES THAT MAY BE LOADED:

- 0093*
- 0094* 0: TREATED AS USER MAIN RELOCATABLE BINARY PROGRAMS
- 0095* 3: USER MAIN RELOCATABLE BINARY PROGRAM
- 0096* 4: DEVICE DRIVER RELOCATABLE BINARY PROGRAMS
- 0097* 5: USER SEGMENT RELOCATABLE BINARY PROGRAMS
- 0098* 6: RELOCATABLE FORTRAN LIBRARY PROGRAMS
- 0099* 7: RELOCATABLE USER BINARY UTILITY PROGRAMS

0100* UPON COMPLETION OF TABLE GENERATION FROM STEP (1) AND (2)

0101* ABOVE, A CHECK IS MADE FOR ANY UNDEFINED EXTERNAL REFERENCES.

0102* IF THERE ARE ANY, THESE MAY BE ENTERED FROM SECONDARY INPUT

0103* DEVICE OR USER FILE, OR USER MAY SPECIFY THAT LOADING IS TO TAKE

0104* PLACE WITH UNDEFINED EXTERNALS.

0105* A SEARCH IS MADE IN USER DIRECTORY FOR MAIN PROGRAM

0106* NAME AND SEGMENT NAMES TO BE LOADED. IF DUPLICATE NAMES OCCUR,

0107* THE LOADER IS ABORTED. THE ABSOLUTE CODE FOR MAIN IS GENERATED

0108* AND PLACED ON DISC. THE USER DIRECTORY IS UPDATED WITH THE NEW

0109* FILE ENTRY AND THE USER/WORK AREA BOUNDARY VALUE IS UPDATED. THE

0110* ABSOLUTE CODE FOR REQUESTED SEGMENTS ARE GENERATED AND FILE NAME

PAGE 0005 #01 DISC OPERATING SYSTEM RELOCATING LOADER

0111* IS ENTERED INTO DIRECTORY,
0112* ERROR DIAGNOSTICS
0113* L01 = CHECKSUM ERROR
0114* L02 = ILLEGAL RECORD
0115* L03 = MEMORY OVERFLOW
0116* L04 = BASE PAGE LINKAGE OVERFLOW
0117* L05 = SYMBOL TABLE OVERFLOW
0118* L06 = MAIN OR SEGMENT NAME ALREADY IN USER OR SYS DIRECTORY
0119* L07 = DUPLICATE ENTRY POINTS
0120* L08 = NO TRANSFER ADDRESS PRESENT
0121* L09 = RECORD OUT OF SEQUENCE
0122* L10 = OUT OF DIRECTORY OR WORK DISC SPACE
0123* L11 = PROGRAM IDENT OVERFLOW
0124* L12 = SPECIFIED FILE NOT IN USER DIRECTORY
0125* L13 = PROGRAM NAME DUPLICATION
0126* L14 = BASE PAGE LENGTH NON-ZERO
0127* L15 = SEGMENT LOADED BEFORE MAIN
0128* L16 = PROGRAM OVERLAY ERROR
0129* L17 = ILLEGAL LIBRARY RECORD
0130* L18 = ILLEGAL BASE PAGE BOUND
0131* L19 = ILLEGAL MAIN PROGRAM BOUND

0133*

0134* PROCESS PROG PARAMETERS

0135*

0136	00000	160001	LOADR	LDA B,I	INPUT UNIT
0137	00001	002002		SZA	
0138	00002	033543R		IOR M300	SET FOR VARIABLE BINARY INP
0139	00003	073416R		STA PGMIN	
0140	00004	006004		INB	
0141	00005	160001		LDA B,I	LIST OUTPUT UNIT
0142	00006	002003		SZA,RSS	
0143	00007	026012R		JMP **3	STANDARD UNIT 6
0144	00010	033441R		IOR M600	SET WRITE BITS
0145	00011	073463R		STA LISTU	
0146	00012	006004		INB	
0147	00013	160001		LDA B,I	NO DEBUG/DEBUG OPTION
0148	00014	073344R		STA DBFLG	
0149	00015	006004		INB	
0150	00016	160001		LDA B,I	CURRENT/BASE PAGE LINK OPTION
0151	00017	073416R		STA CPL	
0152	00020	006004		INB	
0153	00021	160001		LDA B,I	BOUNDS/NO BOUNDS SPECIFICATION
0154	00022	050054		CPA .1	
0155	00023	026654R		JMP STBND	YES - SET PROGRAM BOUNDS
0156	00024	060254		LDA SMFWA	NO - STORE SYSTEM BOUNDS
0157	00025	073460R		STA UMFWA	AS DEFAULT
0158	00026	060100		LDA SMLWA	
0159	00027	073461R		STA UMLWA	
0160	00030	060255		LDA SBFWA	
0161	00031	073456R		STA UBFWA	
0162	00032	060256		LDA SBLWA	
0163	00033	073457R		STA UBLWA	

0164*

0165*

0166* SUPPRESS SYSTEM SEARCH OPTION

0167*

0168	00034	060331	LOAD1	LDA SSFLG	CHECK SYSTEM SEARCH FLAG
0169	00035	053507R		CPA ASCNO	
0170	00036	026043R		JMP LO01	* NO, SO OK
0171	00037	073433R		STA SAVSS	SAVE SS FLAG
0172	00040	002400		CLA	CLEAR SS FLAG IN BASE PAGE
0173	00041	067510R		LDB XSSFL	
0174	00042	016004R		JSB BPSTR	

0175*

0176* SET UP LST, IDENT, & BASE PAGE TABLES

0177*

0178	00043	060147	LO01	LDA EXPG7	GET END BOUND OF LOADER
0179	00044	073511R		STA BPBUF	
0180*					
0181	00045	063456R		LDA UBFWA	FWA OF USER BASE PAGE
0182	00046	073441R		STA TBREL	
0183	00047	073413R		STA PBREL	
0184	00050	003004		CMA,INA	LWA OF USER BASE PAGE BUFFER *
0185	00051	073331R		STA BPOIT	
0186	00052	043457R		ADA UBLWA	# OF USER BASE PAGE AVAILABLE *
0187	00053	073323R		STA BP#	# OF BASE PAGE LINKAGES
0188	00054	043511R		ADA BPBUF	START OF BASE PAGE LINK BUFFER*

0189	00055	073322R	STA BPEND	END OF USER BASE PAGE BUFFER
0190	00056	002004	INA	SET LOADER SYMBOL TABLE ADDRESS
0191	00057	073275R	STA BLST	
0192	00060	073277R	STA PLST	
0193	00061	063511R	LDA BPBUF	
0194	00062	003004	CMA, INA	
0195	00063	043456R	ADA UBFWA	
0196	00064	073407R	STA NBPBF	FORM NEG BASE PAGE BUFFER ADDRESS
0197	00065	060045	LDA N6	SET PROGRAM IDENT TABLE ADDRESS
0198	00066	040100	ADA SMLWA	
0199	00067	073305R	STA BIDNT	
0200	00070	073307R	STA PIDNT	
0201	00071	063323R	LDA BP#	CLEAR BASE PAGE LINKAGE BUFFER
0202	00072	003004	CMA, INA	
0203	00073	073454R	STA WDCNT	SET CLEAR COUNTER
0204	00074	067511R	LDB BPBUF	SET CLEAR BUFFER ADDRESS
0205	00075	002400	CLA	
0206	00076	170001	STA B, I	CLEAR LOCATION
0207	00077	006004	INB	INCREMENT BUFFER ADDRESS
0208	00100	037454R	ISZ WDCNT	SKIP=DONE CLEARING
0209	00101	026076R	JMP *-3	GO CLEAR NEXT LOCATION
0210*				
0211*	VERIFY THAT THERE IS SUFFICIENT DISC AREA &			
0212*	SET INITIAL DISC POINTERS			
0213*				
0214	00102	016001X	JSB EXEC	REQUEST WORK TRACK LIMITS
0215	00103	000110R	DEF **5	
0216	00104	000066	DEF ,17	
0217	00105	011354R	DEF DSKAD	FIRST WORK TRACK ADDRESS
0218	00106	005442R	DEF TR	LAST WORK TRACK ADDRESS
0219	00107	011444R	DEF TEMP	# SECTORS/TRACK ADDRESS
0220	00110	063354R	LDA DSKAD	CHECK IF 3 WORK TRACKS AVAILABLE
0221	00111	003004	CMA, INA	
0222	00112	043442R	ADA TR	
0223	00113	070001	STA B	
0224	00114	040051	ADA N2	
0225	00115	002020	SSA	
0226	00116	026330R	JMP DRCHA	ERROR=NOT 3 WORK TRACKS
0227	00117	063442R	LDA TR	
0228	00120	002004	INA	
0229	00121	001727	ALF, ALF	
0230	00122	072757R	STA DTRTM	FORM LAST LEGAL WORK TRACK+1
0231	00123	063354R	LDA DSKAD	
0232	00124	003004	CMA, INA	
0233	00125	043442R	ADA TR	# OF SCRATCH TRACKS AVAILABLE
0234	00126	001100	ARS	DIVIDE AREA IN HALF
0235	00127	043354R	ADA DSKAD	
0236	00130	001727	ALF, ALF	TRACK ADDRESS
0237	00131	073362R	STA DSKSS	SET BIN SCRATCH
0238	00132	001727	ALF, ALF	RIGHT JUSTIFY
0239	00133	064155	LDB SYSSC	COMPARE SYS SC TO USER SC
0240	00134	054161	CPB CUDSC	
0241	00135	026141R	JMP **4	COMPARE SO USER NOT ON SYS
0242	00136	060154	LDA DISCO	GET LAST TRACK AVAILABLE
0243	00137	010076	AND M3777	
0244	00140	002004	INA	

0245	00141 003004	CMA,INA	MAKE NEG
0246	00142 073363R	STA DSKTM	
0247	00143 060187	LDA UDNTS	USER DISC NEXT AVAILABLE T/S
0248	00144 073353R	STA DSKAB	TRACK/SECTOR ADDRESS
0249*			
0250*	INITIALIZE IDENT & LST TABLES		
0251*	PUT EXEC ENTRY IN LST		
0252*			
0253	00145 016230R	JSB INLST	INITIALIZE LIST
0254	00146 016234R	JSB LSTX	SET CURRENT LST ADDRESSES
0255	00147 000000	NOP	
0256	00150 016261R	JSB INIDX	INITIALIZE PROGRAM IDENT LIST
0257	00151 016265R	JSB IDX	SET CURRENT IDENT ADDRESSES
0258	00152 000000	NOP	
0259	00153 063476R	LDA AEXEC	PUT EXEC INTO LST
0260	00154 173300R	STA LST1,I	
0261	00155 063477R	LDA AEXEC+1	
0262	00156 173301R	STA LST2,I	
0263	00157 063500R	LDA AEXEC+2	
0264	00160 173302R	STA LST3,I	
0265	00161 002400	CLA	
0266	00162 173303R	STA LST4,I	
0267	00163 173304R	STA LST5,I	
0268	00164 063276R	LDA TLST	
0269	00165 073277R	STA PLST	

0271*

0272* PROCESS JBIN RELOCATABLE FILES

0273*

0274	00166	060101	LDA JBINS	CHECK IF ANY JOB BINARY PROGRAMS
0275	00167	002003	SZA, RSS	
0276	00170	026242R	JMP RCNTL	NO
0277	00171	073354R	STA DSKAD	
0278	00172	050102	CPA JBINC	CHECK IF ANY STORED
0279	00173	026242R	JMP RCNTL	NO
0280	00174	060102	LDA JBINC	COMPUTE # SECTORS USED IN CURRNT
0281	00175	010074	AND M377	TRACK
0282	00176	073335R	STA COUNT	
0283	00177	060102	LDA JBINC	
0284	00200	001727	ALF, ALF	
0285	00201	010074	AND M377	
0286	00202	002004	INA	
0287	00203	073442R	STA TR	
0288	00204	010001X	JBN10 JSB EXEC	CHECK IF NEXT TRACK HOLDS BINARY
0289	00205	000212R	DEF +45	DATA
0290	00206	007435R	DEF .16	
0291	00207	000054	DEF .1	# TRACKS
0292	00210	005442R	DEF TR	STARTING TRACK
0293	00211	005441R	DEF SECT	ACTUAL TRACK
0294	00212	063441R	LDA SECT	CHECK IF STARTING = ACTUAL
0295	00213	053442R	CPA TR	
0296	00214	026221R	JMP JBN20	YES-50 BINARY TRACK
0297	00215	002003	SZA, RSS	ZERO INDICATES END OF DISC
0298	00216	026226R	JMP JBN30	NO MORE BINARY TRACKS
0299	00217	073442R	STA TR	
0300	00220	002001	RSS	
0301	00221	037442R	JBN20 ISZ TR	INCREMENT TRACK #
0302	00222	063335R	LDA COUNT	UPDATE CUMULATIVE SECTOR #
0303	00223	040116	ADA SECTR	
0304	00224	073335R	STA COUNT	
0305	00225	026204R	JMP JBN10	GO CHECK NEXT TRACK
0306	00226	002400	JBN30 CLA	READ SYSTEM DISC
0307	00227	073414R	STA PCNTT	
0308	00230	060055	LDA .2	READ SYSTEM DISC
0309	00231	073352R	STA DISCX	
0310	00232	002404	CLA, INA	YES
0311	00233	073443R	STA TDIR	SET JBIN SEARCH FLAG
0312	00234	016362R	JSB SYMBL	SET LST AND IDENT FROM JBIN AREA

0313*

0314* PROCESS USER RELOCATABLE FILES

0315*

0316	00235	002400	RCNTF CLA	SET DISC DIRECTION TO FORWARD
0317	00236	073443R	STA TDIR	
0318	00237	063416R	LDA PGMIN	CHECK IF USER FILES/PAPER TAPE
0319	00240	002003	SZA, RSS	TO BE INPUT
0320	00241	026511R	JMP RCN50	NO, SO GO LOAD
0321	00242	063373R	RCNTL LDA FIOPT	CHECK IF FILE TO BE ENTERED FROM
0322	00243	002003	SZA, RSS	SYSTEM TELETYPE
0323	00244	026251R	JMP RCN02	MAYBE-50 MAKE FURTHER CHECK
0324	00245	060112	LDA SYSTY	YES
0325	00246	033544R	IOR M400	SET ECHO BIT
0326	00247	073444R	STA TEMP	

0327	00250	026261R	JMP	RCN04	GO OUTPUT MESSAGE
0328	00251	060111	RCN02	LDA BATCH	COMPARE BATCH AND SYS TTY DEVICE
0329	00252	050112		CPA SYSTY	
0330	00253	033544R		IOR M400	SET ECHO BIT
0331	00254	073444R		STA TEMP	
0332	00255	060111		LDA BATCH	CHECK IF BATCH IS SYS TELETYPE
0333	00256	050112		CPA SYSTY	
0334	00257	002001		RSS	YES
0335	00260	026267R		JMP RCN05	NO
0336	00261	016001X	RCN04	JSB EXEC	OUTPUT MESSAGE:
0337	00262	000267R		DEF *+5	=ENTER FILE NAME(S) OR /E
0338	00263	000055		DEF .2	
0339	00264	000054		DEF .1	
0340	00265	000346R		DEF CMES	
0341	00266	007433R		DEF .12	
0342	00267	016001X	RCN05	JSB EXEC	REQUEST INPUT FROM BATCH OR SYS
0343	00270	000275R		DEF *+5	
0344	00271	000054		DEF .1	READ REQUEST
0345	00272	011444R		DEF TEMP	CONTROL WORD
0346	00273	011207R	DATB	DEF DATAB	DATA BUFFER ADDRESS
0347	00274	007427R		DEF N72	CHARACTER COUNT
0348	00275	006003		SZ0, RSS	SKIP INPUT IN
0349	00276	026242R		JMP RCNTL	GO TO READ NEXT CONTROL INPUT
0350	00277	007004		CMB, INB	
0351	00300	076626R		STB NEGPM	SAVE NEG INPUT COUNT
0352	00301	062273R		LDA DATB	
0353	00302	000066		CLE, ELA	
0354	00303	072627R		STA PMBUF	SET INPUT BUF CHAR ADDR
0355	00304	063207R		LDA DATAB	CHECK IF INPUT IS /E
0356	00305	053455R		CPA SLSHE	
0357	00306	026511R		JMP RCN50	YES - SO DONE
0358	00307	016472R	RCN10	JSB GETP	GET ONE CONTROL PARAMETER
0359	00310	073445R		STA TERMN	SAVE TERMINATOR FLAG
0360	00311	053455R		CPA SLSHE	
0361	00312	026511R		JMP RCN50	
0362	00313	016001X		JSB EXEC	CHECK IF FILE IN USER DIRECTORY
0363	00314	000320R		DEF *+4	
0364	00315	007436R		DEF .10	
0365	00316	000214R		DEF PBUF	ADDRESS OF FILE NAME
0366	00317	011335R		DEF COUNT	# OF SECTORS IN FILE
0367	00320	073354R		STA DSKAD	SAVE TRACK/SECTOR ADDR OF FILE
0368	00321	002400		CLA	
0369	00322	073443R		STA TDIR	
0370	00323	063335R		LDA COUNT	
0371	00324	002002		SZA	
0372	00325	026334R		JMP RCN30	PRINT UNDEFINED FILE ERROR CODE
0373	00326	063502R		LDA ERR12	
0374	00327	016344R		JSB ERROR	PRINT UNDEFINED FILE NAME
0375	00330	060060		LDA .5	
0376	00331	067563R		LDB DFPBF	
0377	00332	016331R		JSB SYOUT	
0378	00333	026632R		JMP ABORT	ABORT LOADER
0379	00334	016557R	RCN30	JSB PROGL	CHECK IF PROGRAMS SPECIFIED
0380	00335	016000R		JSB SETCR	READ USER DISC
0381	00336	016362R		JSB SYMBL	ENTER ENT/EXT-PROGRAM IDENTIS
0382	00337	063445R		LDA TERMN	GET TERMINATION SYMBOL

0383	00340	053455R	CPA SLSHE	END OF PARAMETERS?
0384	00341	026511R	JMP RCN50	YES
0385	00342	062626R	LDA NEGPM	
0386	00343	002003	SZA, RSS	END OF STATEMENT?
0387	00344	026242R	JMP RCNTL	YES=READ NEXT STATEMENT
0388	00345	026307R	JMP RCN10	GO TO PROCESS NEXT PARAMETER
0389	00346	042516	CMES ASC 12, ENTER	FILE NAME(S) OR /E


```

0391*
0392*   SYMBL ROUTINE READS USER FILE OR JBIN FILE AND ADDS THESE
0393*   PROGRAMS TO THE PROGRAM LIST AND ALL ENT/EXT TO THE LST.
0394*   THE OPTION EXISTS TO ENTER ALL PROGRAMS FROM THE GR JUST
0395*   THOSE SPECIFIED, ALSO IS JBIN AREA, TRACK ADDRESS IS DECREMENTED
0396*   WHEN CROSSING TRACK BOUNDARIES RATHER THAN INCREMENTING.
0397*   CALLING SEQUENCE:
0398*       PRIOR TO ENTERING SET
0399*           DSKAD = STARTING TRACK/SECTOR OF FILE
0400*           COUNT = # SECTORS IN FILE
0401*           PCNTT = 0 IF ALL PROGRAMS IN FILE TO BE LOADED
0402*                 = N IF SPECIFIC PROGRAMS IN FILE TO BE LOADED
0403*           PLIST = PROGRAM NAMES LIST OF PROGRAMS TO BE LOADED
0404*                 (PCNTT NOT ZERO)
0405*           TDIR  = 0 IF USER FILE
0406*                 = 1 IF JBIN AREA
0407*
0408 00362 000000 SYMBL NOP
0409*
0410 00363 062362R   LDA SYMBL   SET UP RETURN ADDRESS FOR BLOCK
0411 00364 016017R   JSB SETUP    ROUTINE
0412 00365 063354R   LDA DSKAD
0413 00366 073543R   STA DSKNM   SET INITIAL NAM RECORD TR/SECT
0414 00367 072104R   STA BLKAD   SET INITIAL DISC ADDRESS
0415*
0416 00370 016032R   JSB BLOCK    INPUT 1ST BLOCK OF CODE
0417*
0418 00371 006400 SYM10 CLB        CLEAR WORDS/PROGRAM
0419 00372 076510R   STB W,PRG    COUNT
0420 00373 016107R   JSB FETCH    GET RECORD TYPE
0421 00374 050054    CPA ,1       NAME RECORD?
0422 00375 002001    RSS          YES
0423 00376 026637R   JMP RCERR
0424*
0425 00377 063414R   LDA PCNTT    CHECK PROGRAMS SPECIFIED FLAG
0426 00400 002003    SZB,RSS      SKIP=YES
0427 00401 026443R   JMP SYM50
0428 00402 003004    CMA,INA
0429 00403 073454R   STA WDCNT    SET # PROG IN PLIST
0430 00404 063564R   LDA PLAD
0431 00405 073321R   STA AMAD     SET SEARCH ADDRESS
0432 00406 067340R SYM20 LDB CURAL "B" POINTING TO
0433 00407 044055    ADB ,2      PROGRAM NAME
0434 00410 163321R   LDA AMAD,I   CHECK IF NAME IN PLIST TABLE
0435 00411 150001    CPA B,I
0436 00412 002001    RSS          NAME 1,2 IN TABLE
0437 00413 026431R   JMP SYM30    NOT CURRENT PROG NAME
0438 00414 037321R   ISZ AMAD
0439 00415 006004    INB
0440 00416 163321R   LDA AMAD,I
0441 00417 150001    CPA B,I
0442 00420 002001    RSS          NAME 3,4 IN TABLE
0443 00421 026432R   JMP SYM30+1  NOT CURRENT PROG NAME.
0444 00422 037321R   ISZ AMAD
0445 00423 006004    INB
0446 00424 160001    LDA B,I

```

PAGE 0013 #01 DOS-III LOADR - (PASS 1) INPUT RELOCATABLES

0447	00425	010075		AND M1774	
0448	00426	153321R		CPA AMAD,I	
0449	00427	026443R		JMP SYM50	PROGRAM IN PLIST TABLE
0450	00430	026433R		JMP SYM30+2	
0451	00431	037321R	SYM30	ISZ AMAD	INCR PLIST ADDRESS
0452	00432	037321R		ISZ AMAD	INCR PLIST ADDRESS
0453	00433	037321R		ISZ AMAD	INCR PLIST ADDRESS
0454	00434	037454R		ISZ WDCNT	INCR PROG COUNT
0455	00435	026406R		JMP SYM20	GO TRY NEXT PROGRAM IN PLIST
0456	00436	016215R	SYM40	JSB SKIP	SKIP OVER REST OF RECORD
0457	00437	016107R		JSB FETCH	
0458	00440	050060		CPA ,5	END RECORD?
0459	00441	026477R		JMP SYM75	
0460	00442	026436R		JMP SYM40	NO=SO SKIP OVER REST OF RECORD
0461*					
0462	00443	017402R	SYM50	JSB NAM	PROCESS NAM RECORD
0463*					
0464	00444	016215R	SYM70	JSB SKIP	SKIP OVER REST OF RECORD
0465	00445	016107R		JSB FETCH	
0466	00446	050055		CPA ,2	ENT RECORD?
0467	00447	026503R		JMP SYENT	PROCESS ENTRY RECORD
0468	00450	050056		CPA ,3	DBL RECORD?
0469	00451	026501R		JMP SYDBL	YES - FIND MAX
0470	00452	050057		CPA ,4	EXT RECORD?
0471	00453	026504R		JMP SYEXT	PROCESS EXTERNAL RECORD
0472	00454	050060		CPA ,5	END RECORD?
0473	00455	002001		RSS	
0474	00456	026637R		JMP RCERR	ILLEGAL RECORD TYPE
0475	00457	017747R		JSB END	PROCESS END RECORD
0476*					
0477	00460	062510R		LDA W,PRG	GET # WORDS IN PROG
0478	00461	040057		ADA ,4	ADD 4 MORE FOR REST OF END
0479	00462	006400		CLB	(FOR DIVIDE)
0480	00463	016002X		DIV ,128	COMPUTE # SECTOS IN PROGRAM
0481	00465	006002		SZB	FROM # WORDS
0482	00466	002004		INA	
0483	00467	072507R		STA #SECT	
0484*					
0485	00470	163317R		LDA ID7,I	STORE #TRACKS OR MAIN SEG
0486	00471	002003		SZA,RSS	
0487	00472	062507R		LDA #SECT	GET CURRENT PROG SECTOR COUNT
0488	00473	067443R		LDB TDIR	FORWARD OR BACKWARD DIRECTION?
0489	00474	006002		SZB	FORWARD
0490	00475	003004		CMA,INA	BACKWARD=COMPLEMENT # SECTORS
0491	00476	173317R		STA ID7,I	
0492*					
0493	00477	016215R	SYM75	JSB SKIP	SKIP REST OF END RECORD
0494	00500	026371R		JMP SYM10	PROCESS NEXT PROGRAM
0495*					
0496*					
0497	00501	017544R	SYDBL	JSB DBL	PROCESS DBL RECORD
0498	00502	026444R		JMP SYM70	
0499*					
0500	00503	003401	SYENT	CCA,RSS	ENT
0501	00504	002400	SYEXT	CLA	EXT
0502	00505	017565R		JSB EN,XT	PROCESS ENT/EXT RECORDS

0503 00506 026444R JMP SYM70

0504*

0505 00507 000000 #SECT NOP

SECTORS/PROGRAM

0506 00510 000000 W.PRG NOP

WORDS/PROGRAM

```

0508*
0509*  PROCESS RELOCATBLE BINARY INPUT
0510*
0511  00511 060055 RCN50 LDA .2      SET FOR SYSTEM DISC
0512  00512 073352R      STA DISCX
0513*
0514  00513 063362R      LDA DSKSS    SET SCRATCH AREA AS DISC ADDRESS
0515  00514 073354R      STA DSKAD    FOR OUTPUTTING RELOCATABLES
0516*
0517  00515 063416R      LDA PGMIN    CHECK IF ANY INPUT FROM SECONDAR
0518  00516 002003      SZA,RSS      INPUT DEVICE
0519  00517 026761R      JMP LODLB    NO-GO CHECK LIBRARY PROGRAM
0520  00520 010072      AND M77
0521  00521 050055      CPA .2
0522  00522 026761R      JMP LODLB    NO PAPER TAPE TO BE LOADED
0523*
0524*  READ IN BINARY RELOCATABLE INPUT FROM SECONDARY DEVICE AND
0525*  STORE IN PACKED FORMAT ONTO DISC.
0526*  NAM RECORDS ARE ENTERED INTO PROG NAME NAME TABLE AND ENT/EXT
0527*  ARE ENTERED INTO LOADER SYMBOL TABLE.
0528*
0529  00523 003400 RCN60 CCA
0530  00524 073420R      STA PLFLG    SET LOADING FLAG
0531*
0532  00525 063547R      LDA N128
0533  00526 073350R      STA DCNT     INITIALIZE DBUF COUNT
0534  00527 066774R      LDB AABUF
0535  00530 077336R      STB CURAD    INITIALIZE DBUF POINTER
0536  00531 016516R      JSB BUFCL    CLEAR DBUF
0537  00532 002400      CLA           SET FOR 1ST OUTPUT BUFFER
0538  00533 072751R      STA ABFLG
0539  00534 072450R      STA FIRST
0540*
0541  00535 063307R      LDA PIDNT
0542  00536 072657R      STA SIDNT    OF CHECKSUM ERROR
0543  00537 063277R      LDA PLST     SAVE END OF LST TABLE
0544  00540 072656R      STA SLST
0545  00541 063354R      LDA DSKAD
0546  00542 072655R      STA SDSK
0547*
0548  00543 060065 TREAD LDA .10      PRINT -LOAD TAPE- ON SYS TTY
0549  00544 066660R      LDB MESLD
0550  00545 016331R      JSB SYOUT
0551  00546 016001X      JSB EXEC     SUSPEND PROGRAM
0552  00547 000551R      DEF ++2
0553  00550 000062      DEF .7
0554  00551 160001      LDA B,I
0555  00552 002002      SZA
0556  00553 026761R      JMP LODLB    GO TO CHECK IF LOAD LIBRARY
0557  00554 003400      CCA           SET EOT FLAG TO IGNORE EOT
0558  00555 073370R      STA ETFLG
0559*
0560  00556 016001X LDRIN JSB EXEC     READ INPUT FROM 2NDARY INPUT DEV
0561  00557 000564R      DEF ++5      DEVICE INTO LBUF
0562  00560 000054      DEF .1
0563  00561 011416R      DEF PGMIN

```

0564	00562	010207R	DEF LBUF	
0565	00563	007550R	DEF ,128	
0566*				
0567	00564	012566R	AND M240	
0568	00565	002302	CCE, SZA	END OF FILE/TAPE?
0569	00566	000240	M240 OCT 240	=YES (CLE INSTRUCTION)
0570	00567	006042	SEZ, SZB	REC LENGTH=0?
0571	00570	026575R	JMP TESTR	YES, SO GO PROCESS RECORD
0572	00571	063370R	LDA ETFLG	NO, SO CHECK END OF TAPE
0573	00572	002002	SZA	
0574	00573	026556R	JMP LDRIN	GO READ IN NEXT RECORD
0575	00574	026543R	JMP TREAD	OUTPUT =LOAD= AND READ IN RECORD
0576*				
0577	00575	002400	TESTR CLA	SET EOT FLAG = VALID EOT
0578	00576	073370R	STA ETFLG	
0579*				
0580	00577	067553R	LDB ALBUF	
0581	00600	077340R	STB CURAL	INITIALIZE LBUF POINTER
0582	00601	063547R	LDA N128	
0583	00602	073403R	STA LCNT	INITIALIZE LBUF COUNT
0584*				
0585	00603	016107R	JSB FETCH	GET RECORD TYPE, CHK VALIDITY
0586*				
0587	00604	073410R	STA TEMP1	SAVE RECORD TYPE
0588	00605	163340R	LDA CURAL, I	COMPUTE CHECKSUM
0589	00606	067340R	LDB CURAL	
0590	00607	244055	AOB ,2	
0591	00610	140001	ADA B, I	
0592	00611	006004	INB	
0593	00612	037454R	ISZ WDCNT	
0594	00613	026610R	JMP *-3	
0595*				
0596	00614	067340R	LDB CURAL	COMPUTED = ACTUAL CHECKSUM?
0597	00615	006004	INB	
0598	00616	150001	CPA B, I	
0599	00617	002001	RSS	YES =
0600	00620	026643R	JMP CHKER	= CHECKSUM ERROR
0601	00621	063410R	LDA TEMP1	RESTORE RECORD TYPE
0602*				
0603	00622	067420R	LDB PLFLG	
0604	00623	050054	CPA ,1	TYPE = NAM RECORD ?
0605	00624	026666R	JMP NAMR	YES
0606	00625	006002	SZB	
0607	00626	026641R	JMP NMERR	RECORD OUT OF SEQUENCE
0608	00627	050055	CPA ,2	TYPE = ENT RECORD ?
0609	00630	026703R	JMP ENTR	YES
0610	00631	050056	CPA ,3	TYPE=DL RECORD?
0611	00632	026701R	JMP DBLR	YES
0612	00633	050057	CPA ,4	TYPE = EXT RECORD ?
0613	00634	026704R	JMP EXTR	YES
0614	00635	050060	CPA ,5	
0615	00636	026707R	JMP ENDR	
0616*				
0617	00637	063470R	RCERR LDA ERR02	ILLEGAL RECORD ERROR.
0618	00640	026631R	JMP ABRTX	
0619*				

0620	00641	063477R	NMERR	LDA	ERR09	RECORD OUT OF SEQUENCE
0621	00642	026631R		JMP	ABRTX	
0622*						
0623	00643	063467R	CHKER	LDA	ERR01	
0624	00644	016344R		JSB	ERROR	OUTPUT LO1 MESSAGE
0625	00645	062657R		LDA	SIDNT	RESTORE IDENT & LST TABLES TO
0626	00646	073307R		STA	PIDNT	THEIR INITIAL STATE
0627	00647	062656R		LDA	SLST	
0628	00650	073277R		STA	PLST	
0629	00651	062655R		LDA	SDSK	
0630	00652	073354R		STA	DSKAD	
0631	00653	016636R		JSB	SET	SET SIMULATED END-OF-TAPE
0632	00654	026523R		JMP	RCN60	TRY ENTERING PROGRAM AGAIN
0633*						
0634	00655	000000	SDSK	NOP		SAVED DISC ADDRESS
0635	00656	000000	SLST	NOP		SAVED END OF LST TABLE
0636	00657	000000	SIDNT	NOP		SAVED END OF IDENT TABLE
0637	00660	000661R	MESLD	DEF	**+1	
0638	00661	046117		ASC	5,LOAD	TAPE
0640	00666	006003	NAMR	SZB,	RSS	
0641	00667	026641R		JMP	NMERR	RECORD OUT OF SEQUENCE
0642*						
0643	00670	006400		CLB		
0644	00671	077367R		STB	DSCNT	CLEAR DISC SEGMENT COUNT
0645	00672	077334R		STB	CNFLG	CLEAR DISC SEGMENT COUNT FLAG
0646	00673	077420R		STB	PLFLG	SET PLFLG = NOT LOADING
0647	00674	063354R		LDA	DSKAD	SET DISC ADDRESS OF NAM RECORD
0001	00675	073543R		STA	DSKNM	FOR DIRECTORY ENTRY,
0002*						
0003	00676	017402R		JSB	NAM	PROCESS NAM RECORD
0004*						
0005	00677	016715R	OUTR	JSB	DWRIT	PACK RECORD, OUTPUT TO DISC
0006	00700	026556R		JMP	LDRIN	GO READ NEXT RECORD
0008	00701	017544R	DBLR	JSB	DBL	PROCESS DBL RECORD
0009	00702	026677R		JMP	OUTR	OUTPUT DBL RECORD TO DISC
0011	00703	003401	ENTR	CCA,	RSS	
0012	00704	002400	EXTR	CLA		
0013	00705	017565R		JSB	EN,XT	PROCESS ENT/EXT RECORDS
0014	00706	026677R		JMP	OUTR	OUTPUT RECORD TO DISC
0016	00707	003400	ENDR	CCA		
0017	00710	073420R		STA	PLFLG	
0018	00711	073334R		STA	CNFLG	
0019	00712	016715R		JSB	DWRIT	WRITE END RECORD ON DISC
0020*						
0021	00713	017747R		JSB	END	PROCESS END RECORD
0022	00714	026556R		JMP	LDRIN	

```

0024*
0025*      DWRIT PACKS THE CONTENTS OF LBUF INTO DBUF AND DUMPS
0026*      DBUF WHEN IT CONTAINS 128 WORDS OF RELOCATABLE INPUT.
0027*
0028 00715 000000 DWRIT NOP
0029 00716 067553R LDB ALBUF      GET ADDRESS OF LBUF
0030 00717 076760R STB LTEMP
0031 00720 062207R LDA LBUF      GET RECORD LENGTH
0032 00721 001727 ALF,ALF
0033 00722 003004 CMA,INA
0034 00723 073403R STA LCNT
0035 00724 162760R LTOD LDA LTEMP,I  GET WORD FROM LBUF AND
0036 00725 173336R STA CURAD,I  SET WORD INTO DBUF
0037 00726 037350R ISZ DCNT
0038 00727 026732R JMP GETL
0039 00730 016377R JSB DDOUT    OUTPUT IF BUFFER FULL
0040 00731 002001 RSS
0041 00732 037336R GETL ISZ CURAD    INCR DBUF ADDRESS
0042 00733 036760R ISZ LTEMP    INCR LBUF ADDRESS
0043 00734 037403R ISZ LCNT
0044 00735 026724R JMP LTOD     GO MOVE NEXT WORD
0045 00736 063334R LDA CNFLG    TEST END FLAG
0046 00737 002003 SZA,RSS      SKIP IF END RECORD READ
0047 00740 026753R JMP MAXWK    CHECK FOR OVERFLOW
0048 00741 063350R LDA DCNT     CHECK IF ANY DATA IN BUFFER
0049 00742 053547R CPA N128
0050 00743 002001 RSS
0051 00744 016377R JSB DDOUT    OUTPUT END RECORD TO DISC
0052 00745 163316R LDA ID6,I    CHECK PROGRAM TYKPE
0053 00746 010073 AND M177
0054 00747 050060 CPA .5
0055 00750 026753R JMP MAXWK    USER SEGMENT - SO CHECK FOR OVFL
0056 00751 063367R LDA DSCNT
0057 00752 173317R STA ID7,I    STORE SECTOR COUNT IN IDENT
0058*
0059*      CHECK IF RELOCATABLE BINARY EXCEEDS AVAILABLE WORK TRACK SPACE
0060*
0061 00753 062757R MAXWK LDA DTRTM    MAX + 1
0062 00754 053354R CPA DSKAD    CURRENT
0063 00755 026330R JMP DRCHA    ERROR-OUT OF WORK TRACKS
0064 00756 126715R JMP DWRIT,I  RETURN
0065*
0066 00757 000000 DTRTM NOP
0067 00760 000000 LTEMP NOP

```

```

0069*
0070*  UPDATE LST, CHECK FOR UNDEFINED EXT &
0071*  LOAD LIBRARY ROUTINES IF NECESSARY
0072*
0073  00761 063344R LODLB LDA DBFLG
0074  00762 002003      SZA,RSS      ENTER DEBUG INTO LST
0075  00763 027010R      JMP NODBG      NO DEBUG OPTION
0076  00764 063345R      LDA DBFL2      HAS DEBUG BEEN ENTERED IN LST?
0077  00765 002002      SZA
0078  00766 027010R      JMP NODBG      YES- DO NOT RE-ENTER.
0079  00767 016230R      JSB INLST
0080  00770 016234R      JSB LSTX
0081  00771 026773R      JMP ++2
0082  00772 026770R      JMP --2
0083  00773 063462R      LDA CHRDE      SET -DEBUG- NAME INTO LST AND ID
0084  00774 173300R      STA LST1,I
0085  00775 063463R      LDA CHRDE+1
0086  00776 173301R      STA LST2,I
0087  00777 063464R      LDA CHRDE+2
0088  01000 010075      AND M1774
0089  01001 173302R      STA LST3,I
0090  01002 002400      CLA      CLEAR BASE PAGE LINK ADDRESS
0091  01003 173303R      STA LST4,I
0092  01004 173304R      STA LST5,I
0093  01005 063276R      LDA TLST
0094  01006 073277R      STA PLST
0095  01007 073345R      STA DBFL2      SET FOR DEBUG ALREADY IN LST
0096  01010 002400      NODBG CLA
0097  01011 073371R      STA EXCNT
0098  01012 016230R      LOADL JSB INLST
0099  01013 016234R      JSB LSTX
0100  01014 027061R      JMP LODLL
0101  01015 063476R      LDA AEXEC
0102  01016 153300R      CPA LST1,I
0103  01017 002001      RSS
0104  01020 027043R      JMP LODLX
0105  01021 063477R      LDA AEXEC+1
0106  01022 153301R      CPA LST2,I
0107  01023 002001      RSS
0108  01024 027043R      JMP LODLX
0109  01025 163302R      LDA LST3,I
0110  01026 010075      AND M1774
0111  01027 053500R      CPA AEXEC+2
0112  01030 002001      RSS
0113  01031 027043R      JMP LODLX
0114  01032 002400      CLA
0115  01033 173303R      STA LST4,I
0116  01034 063107R      LDA GOCHK      GET LINK FOR EXEC EXT
0117  01035 013444R      AND 01777
0118  01036 173304R      STA LST5,I
0119  01037 163302R      LDA LST3,I
0120  01040 010075      AND M1774
0121  01041 173302R      STA LST3,I
0122  01042 027013R      JMP LOADL+1
0123  01043 002400      LODLX CLA
0124  01044 173304R      STA LST5,I

```


0125	01045	163302R	LDA	LST3,I	
0126	01046	010075	AND	M1774	
0127	01047	173302R	STA	LST3,I	
0128	01050	163303R	LDA	LST4,I	
0129	01051	002020	SSA		
0130	01052	027055R	JMP	++3	
0131	01053	002002	SZA		
0132	01054	027013R	JMP	LOADL+1	
0133	01055	037371R	ISZ	EXCNT	
0134	01056	003400	CCA		
0135	01057	173303R	STA	LST4,I	
0136	01060	027013R	JMP	LOADL+1	
0137	01061	063371R	LODLL	LDA EXCNT	
0138	01062	002003	SZA	RSS	
0139	01063	026226R	JMP	LMAIN	
0140	01064	016230R	JSB	INL9T	INIT, LST ENTRY
0141	01065	016234R	JSB	LSTX	GET NEXT LST
0142	01066	026226R	JMP	LMAIN	NO UNDEFINES: GO LOAD
0143	01067	163303R	LDA	LST4,I	CHECK IF UNDEFINED EXT
0144	01070	050052	CPA	N1	
0145	01071	002001	RSS		YES: SEARCH LIBRARY
0146	01072	027065R	JMP	+-5	TRY NEXT LST
0147	01073	002400	CLA		
0148	01074	073371R	STA	EXCNT	
0149	01075	073367R	STA	DSCNT	
0150	01076	060055	LDA	.2	SET FOR SYSTEM DISC
0151	01077	073352R	STA	DISCX	
0152	01100	017133R	JSB	LIBS	LOAD REQUESTED LIBRARY ROUTINES
0153	01101	063371R	LDA	EXCNT	GET # LIB ROUTINES LOADED
0154	01102	002002	SZA		NONE, SO DONE LOADING ROUTINES
0155	01103	027061R	JMP	LODLL	
0156	01104	016630R	JSB	LSTEX	LIST UNDEFINED EXTERNALS
0157	01105	002001	RSS		UNDEFINED EXT FOUND
0158	01106	026226R	JMP	LMAIN	NO UNDEFINED EXT FOUND
0159	01107	016001X	GOCHK	JSB EXEC	SUSPEND LOADER
0160	01110	001112R	DEF	++2	
0161	01111	000062	DEF	.7	
0162	01112	160001	LDA	B,I	CHECK IF :GO,P,N
0163	01113	002002	SZA		
0164	01114	027124R	JMP	CFILE	
0165	01115	006004	INB		
0166	01116	160001	LDA	B,I	CHECK IF N SPECIFIED
0167	01117	002003	SZA	RSS	
0168	01120	026523R	JMP	RCN60	GO INPUT FROM SECONDARY DEVICE
0169	01121	033543R	IOR	M300	
0170	01122	073416R	STA	PGMIN	SET INPUT LOGICAL UNIT
0171	01123	026523R	JMP	RCN60	GO INPUT FROM SECONDARY DEVICE
0172	01124	050055	CFILE	CPA .2	CHECK IF MORE FILES TO BE INPUT
0173	01125	027131R	JMP	CF10	YES
0174	01126	050054	CPA	.1	CHECK IF TO IGNORE UNDEFINES
0175	01127	026226R	JMP	LMAIN	YES
0176	01130	027107R	JMP	GOCHK	GET :GO INPUT
0177	01131	037373R	CF10	ISZ FIOPT	
0178	01132	026242R	JMP	RCNTL	
0179*					
0180*					

LIBS SEARCHES FORTRAN RELOCATABLE LIBRARY FOR REFERENCED

```

0181* PROGRAMS NOT YET SATISFIED BY OTHER RELOCATABLE BINARY INPUTS.
0182*
0183 01133 000000 LIBS NOP
0184 01134 063133R LDA LIBS SET RETURN ADDRESS FOR BLOCK
0185 01135 016017R JSB SETUP ROUTINE
0186*
0187 01136 003400 CCA
0188 01137 073377R STA IGNOR
0189 01140 060325 LDA DSCL# SET # OF
0190 01141 003004 CMA,INA LIBRARY ROUTINES
0191 01142 073340R STA #LIB
0192*
0193 01143 063446R LDA M0760 SET MAX # LIBRARY SECTORS
0194 01144 073335R STA COUNT TO 76000B
0195 01145 060324 LDA DSCLB SET INITIAL LIBRARY TRACK/SECTOR
0196 01146 072104R STA BLKAD DISC ADDRESS
0197 01147 073543R STA DSKNM SET INITIAL NAM RECORD TR/SECT
0198*
0199 01150 016032R LIRD JSB BLOCK INPUT 1ST BLOCK OF RELOCATABLES
0200*
0201 01151 016107R LIB1 JSB FETCH GET RECORD TYPE
0202 01152 050054 CPA .1 CHECK IF NAME RECORD
0203 01153 002001 RSS
0204 01154 027202R JMP LBERR INVALID LIBRARY RECORD
0205 01155 016230R JSB INLST CHECK IF UNDEFINES PRESENT
0206 01156 016234R JSB LSTX
0207 01157 026100R JMP BLCK6
0208 01160 163303R LDA LST4,I CHECK LST4 FOR -1
0209 01161 050052 CPA N1
0210 01162 002001 RSS SKIP--MORE UNDEFINES PRESENT
0211 01163 027156R JMP *-5 LST DEFINED--TRY NEXT LST
0212 01164 063377R LDA IGNOR CHECK IF LOADING
0213 01165 002002 SZA YES
0214 01166 027170R JMP LISKP GO SKIP REST OF RECORD
0215 01167 017402R JSB NAM PROCESS NAM RECORD
0216 01170 016215R LISKP JSB SKIP SKIP REST OF RECORD
0217 01171 016107R JSB FETCH GET NEXT RECORD TYPE
0218 01172 050055 LINNM CPA .2 ENT RECORD?
0219 01173 027211R JMP LENTR YES--SO PROCESS ENT RECORD
0220 01174 050056 CPA .3 DBL RECORD?
0221 01175 027205R JMP LDBLR YES--SO PROCESS DBL RECORD
0222 01176 050057 CPA .4 EXT RECORD?
0223 01177 027212R JMP LEXTR YES--SO PROCESS EXT RECORD
0224 01200 050060 CPA .5 END RECORD?
0225 01201 027316R JMP LENDR YES--SO PROCESS END RECORD
0226 01202 063204R LBERR LDA ERR17
0227 01203 026631R JMP ABRTX
0228*
0229 01204 030467 ERR17 ASC 1,17 BAD LIBRARY
0230*
0231 01205 063377R LDBLR LDA IGNOR LOADING?
0232 01206 002003 SZA,RSS
0233 01207 017544R JSB DBL YES - PROCESS DBL RECORD
0234 01210 027170R JMP LISKP NO - SKIP REST OF RECORD
0235*
0236 01211 003401 LENTR CCA,RSS

```

0237	01212	002400	LEXTR	CLA	
0238	01213	073410R		STA NXFLG	SET ENT/EXT FLAG
0239	01214	163340R		LDA CURAL,I	GET # ENTRIES IN ENT/EXT
0240	01215	010071		AND M37	
0241	01216	003004		CMA,INA	
0242	01217	073445R		STA TERMN	SET SYMBOL COUNT
0243	01220	063377R		LDA IGNOR	GET UTILITY PROGRAM LOADING FLAG
0244	01221	002003		SZA,RSS	NOT LOADING
0245	01222	027226R		JMP PRENX	PROCESS ENT/EXT RECORD
0246	01223	063410R		LDA NXFLG	GET ENT/EXT FLAG
0247	01224	002003		SZA,RSS	SKIP-ENT
0248	01225	027170R		JMP LISKP	IGNOR EXT RECORD
0249	01226	016203R	PRENX	JSB BUMP	INCREMENT TO
0250	01227	016203R		JSB BUMP	NAME ADDRESS
0251	01230	163340R	NSYMB	LDA CURAL,I	GET NAME
0252	01231	072210R		STA TBUF	
0253	01232	016203R		JSB BUMP	
0254	01233	163340R		LDA CURAL,I	
0255	01234	072211R		STA TBUF+1	
0256	01235	016203R		JSB BUMP	
0257	01236	163340R		LDA CURAL,I	
0258	01237	072212R		STA TBUF+2	
0259	01240	016203R		JSB BUMP	
0260	01241	017341R		JSB LSCAN	SCAN LST FOR NAME
0261	01242	027257R		JMP ENEX2	NAME NOT IN LST
0262	01243	063377R		LDA IGNOR	GET UTILITY LOADING FLAG
0263	01244	002002		SZA	LOADING
0264	01245	027306R		JMP SCANX	NOT LOADING
0265	01246	063410R		LDA NXFLG	GET ENT/EXT FLAG
0266	01247	002003		SZA,RSS	PROCESS ENT
0267	01250	027267R		JMP ENEX3	SET EXT ORDINAL
0268	01251	163303R		LDA LST4,I	GET ENT/EXT FOR LST ENTRY
0269	01252	002003		SZA,RSS	SKIP-UNDEFINED EXT
0270	01253	027271R		JMP EXNND	IGNORE UTILITY ENTRY POINT
0271	01254	063311R	ENXX1	LDA ID1	
0272	01255	173303R		STA LST4,I	SET LST4=LIBRARY ENT
0273	01256	027271R		JMP EXNND	TEST FOR END OF RECORD
0274	01257	063410R	ENEX2	LDA NXFLG	GET ENT/EXT FLAG
0275	01260	002002		SZA	
0276	01261	027277R		JMP ENEX4	PROCESS ENTRY POINT
0277	01262	017367R		JSB SELST	MAKE LST ENTRY
0278	01263	003400		CCA	
0279	01264	173303R		STA LST4,I	SET LST4 TO UNDEF EXT
0280	01265	002400		CLA	
0281	01266	173304R		STA LST5,I	CLEAR BP LINK ADDRESS WORD
0282	01267	062212R	ENEX3	LDA TBUF+2	
0283	01270	173302R		STA LST3,I	SET ORDINAL INTO LST
0284	01271	063410R	EXNND	LDA NXFLG	GET ENT/EXT FLAG
0285	01272	002002		SZA	EXT
0286	01273	016203R		JSB BUMP	
0287	01274	037445R		ISZ TERMN	SKIP-ALL SYMBOLS PROCESSED
0288	01275	027230R		JMP NSYMB	PROCESS NEXT SYMBOL
0289	01276	027170R		JMP LISKP	GO TO NEXT RECORD
0290	01277	063377R	ENEX4	LDA IGNOR	GET LOADING FLAG
0291	01300	002002		SZA	SKIP-LOADING
0292	01301	027271R		JMP EXNND	IGNORE-ENT NOT IN LIST

0293	01302	017367R		JSB SELST	SET NAME INTO LST
0294	01303	002400		CLA	
0295	01304	173304R		STA LST5,I	CLEAR BP LINK ADDRESS
0296	01305	027254R		JMP ENXX1	CONTINUE PROCESSING ENT POINT
0297	01306	163303R	SCANX	LDA LST4,I	GET ENT/EXT ENTRY FLAG
0298	01307	002021		SSA,RSS	SKIP-UNDEFINED EXT
0299	01310	027271R		JMP EXNND	IGNORE DEFINED EXT
0300	01311	063543R		LDA DSKNM	RESTORE DISC ADDRESS TO START
0301	01312	072104R		STA BLKAD	OF NAM RECORD
0302	01313	002400		CLA	
0303	01314	073377R		STA IGNOR	
0304	01315	027180R		JMP LIRD	GO TO LOAD UTILITY PROGRAM
0305*					
0306	01316	063377R	LENDR	LDA IGNOR	GET LIBRARY LOADING FLAG
0307	01317	002002		SZA	SKIP-LOADING
0308	01320	027325R		JMP OMITE	SKIP OVER END RECORD
0309	01321	017787R		JSB UPNAM	UPDATE NAM RECORD IF NECESSARY
0310	01322	003400		CCA	SET UTIL LOADING FLAG = IGNOR
0311	01323	073377R		STA IGNOR	
0312	01324	037371R		ISZ EXCNT	INCREMENT # UTIL PROG LOADED FLG
0313	01325	016215R	OMITE	JSB SKIP	SKIP OVER END RECORD
0314	01326	016230R		JSB INLST	INITIALIZE LST
0315	01327	016234R	CLODR	JSB LSTX	SET LST ADDRESSES
0316	01330	027335R		JMP END1	TEST FOR DONE
0317	01331	163302R		LDA LST3,I	
0318	01332	010075		AND M1774	
0319	01333	173302R		STA LST3,I	
0320	01334	027327R		JMP CLODR	GO CLEAR NEXT
0321*					
0322	01335	037340R	END1	ISZ #LIB	PROCESSED ALL LIB ROUTINES?
0323	01336	027151R		JMP LIB1	NO- GET NEXT ONE
0324	01337	026100R		JMP BLCK6	YES- RETURN
0325*					
0326	01340	000000	#LIB	NOP	
0327*					
0328*					LSCAN SEARCHES FOR AN ENTRY IN LST IDENTICAL
0329*					TO NAME IN TBUF.
0330*					
0331*					RETURN;
0332*					(N+1)-END OF LST
0333*					(N+2)-NAME FOUND IN LST. CURRENT LST ADDRESSES
0334*					POINT TO THIS ENTRY
0335*					
0336	01341	000000	LSCAN	NOP	
0337	01342	016230R		JSB INLST	INITIALIZE LST
0338	01343	016234R	ENTXX	JSB LSTX	SET LST ADDRESSES
0339	01344	127341R		JMP LSCAN,I	END OF LST RETURN
0340	01345	062210R		LDA TBUF	CHECK IF NAME 1,2 SAME?
0341	01346	153300R		CPA LST1,I	
0342	01347	002001		RSS	YES
0343	01350	027343R		JMP ENTXX	NO-TRY NEXT ENTRY
0344	01351	062211R		LDA TBUF+1	CHECK IF NAME 3,4 SAME?
0345	01352	153301R		CPA LST2,I	
0346	01353	002001		RSS	YES
0347	01354	027343R		JMP ENTXX	NO-TRY NEXT ENTRY
0348	01355	163302R		LDA LST3,I	CHECK IF NAME 5 SAME

```

0349 01356 010075      AND  M1774
0350 01357 072213R     STA  TBUF+3
0351 01360 062212R     LDA  TBUF+2
0352 01361 010075      AND  M1774
0353 01362 062213R     CPA  TBUF+3
0354 01363 002001      RSS
0355 01364 027343R     JMP  ENTXX      YES
0356 01366 037341R     ISZ  LSCAN     NO=TRY NEXT ENTRY
0357 01366 127341R     JMP  LSCAN,I    INCR EXIT ADDRESS
0358*                                     (N+2)RETURN
0359*      SELST SETS CURRENT NAME INTO LST
0360*
0361 01367 000000      SELST NOP
0362 01370 062210R     LDA  TBUF      STORE NAME INTO LST
0363 01371 173300R     STA  LST1,I
0364 01372 062211R     LDA  TBUF+1
0365 01373 173301R     STA  LST2,I
0366 01374 062212R     LDA  TBUF+2
0367 01375 010075      AND  M1774
0368 01376 173302R     STA  LST3,I
0369 01377 063276R     LDA  TLST
0370 01400 073277R     STA  PLST      SET NEW END OF LST
0371 01401 127367R     JMP  SELST,I    RETURN

```

```

0373*
0374* THIS ROUTINE PROCESSES RELOCATABLE NAM RECORDS.
0375* IT IS ASSUMED THAT "FETCH" HAS BEEN CALLED AND
0376* HAS RETURNED A RECORD TYPE 1, IN ADDITION, THE
0377* BUFFER POINTER IS POINTING TO THE 2ND WORD IN
0378* THE RECORD.
0379*
0380* THREE VARIABLES MUST HAVE PRE-FILLED VALUES:
0381*
0382*     DISCX - "2" FOR SYSTEM DISC
0383*           "3" FOR USER DISC
0384*
0385*     DSKNM - DISC ADDRESS OF NAM RECORD
0386*
0387*     RLGTH - RECORD LENGTH (FROM FETCH ROUTINE)
0388*
0389* FUNCTIONS:
0390*
0391*     1. CHECKS TO SEE IF NAME IS ALREADY IN IDENT
0392*        TABLE
0393*
0394*         A. IF SO, ABORTS WITH ERROR 13.
0395*
0396*         B. IF NOT, CREATES AN IDENT ENTRY AS FOLLOWS:
0397*
0398*             ID1 - CHARACTERS 1&2 OF NAME
0399*             ID2 - CHARACTERS 3&4 OF NAME
0400*             ID3 - CHARACTER 5 OF NAME
0401*             ID4 - COMMON LENGTH (BIT 15 SET IF
0402*                  FROM USER DISC)
0403*             ID5 - DISC ADDRESS
0404*             ID6 - PROGRAM TYPE
0405*             ID7 - ZERO
0406*
0407*     2. CHECKS TO SEE IF PROGRAM HAS ANY BASE PAGE
0408*        ALLOCATED
0409*
0410*         A. IF SO, ABORTS WITH ERROR 14.
0411*
0412*         B. IF NOT, RETURNS NORMALLY.
0413*
0414* 01402 000000 NAM NOP
0415*
0416* GET INFORMATION FROM NAM RECORD
0417*
0418* 01403 016203R JSB BUMP
0419* 01404 016203R JSB BUMP
0420* 01405 163340R LDA CURAL,I NAME,
0421* 01406 073534R STA NAM12 CHARACTERS 1&2
0422* 01407 016203R JSB BUMP
0423* 01410 163340R LDA CURAL,I NAME,
0424* 01411 073535R STA NAM34 CHARACTERS 3&4
0425* 01412 016203R JSB BUMP
0426* 01413 163340R LDA CURAL,I NAME-
0427* 01414 010075 AND M1774
0428* 01415 073536R STA NAM5 CHARACTER 5

```

0429	01416	016203R	JSB BUMP	
0430	01417	163340R	LDA CURAL,I	
0431	01420	073537R	STA NPL	PROGRAM LENGTH
0432	01421	016203R	JSB BUMP	
0433	01422	163340R	LDA CURAL,I	
0434	01423	073540R	STA NBPL	BASE PAGE LENGTH
0435	01424	016203R	JSB BUMP	
0436	01425	163340R	LDA CURAL,I	
0437	01426	073541R	STA NCL	COMMON LENGTH
0438	01427	016203R	JSB BUMP	
0439	01430	163340R	LDA CURAL,I	
0440	01431	073542R	STA NTYPE	PROGRAM TYPE
0441*				
0442*	SEARCH FOR DUPLICATE IDENT ENTRY			
0443*				
0444	01432	016261R	JSB INIDX	
0445	01433	016265R	NXTID JSB IDX	
0446	01434	027464R	JMP NTFND	
0447*				
0448	01435	063377R	LDA IGNOR	IGNORE CHECK OF DUPLICATE
0449	01436	002003	SZA,RSS	NAMES IF LOADING FROM
0450	01437	027433R	JMP NXTID	LIBRARY
0451*				
0452	01440	063534R	LDA NAM12	
0453	01441	153311R	CPA ID1,I	
0454	01442	002001	RSS	
0455	01443	027433R	JMP NXTID	
0456	01444	063535R	LDA NAM34	
0457	01445	153312R	CPA ID2,I	
0458	01446	002001	RSS	
0459	01447	027433R	JMP NXTID	
0460	01450	067536R	LDB NAM5	
0461	01451	163313R	LDA ID3,I	
0462	01452	010075	AND M1774	
0463	01453	050001	CPA B	
0464	01454	002001	RSS	
0465	01455	027433R	JMP NXTID	
0466*				
0467	01456	063503R	LDA ERR13	DUPLICATE PROGRAM NAME
0468	01457	016344R	JSB ERROR	
0469	01460	060060	LDA .5	
0470	01461	067311R	LDB ID1	
0471	01462	016331R	JSB SYOUT	
0472	01463	026632R	JMP ABORT	-ERROR ABORT
0473*				
0474*	CREATE IDENT ENTRY			
0475*				
0476	01464	063534R	NTFND LDA NAM12	INSERT PROGRAM NAME
0477	01465	173311R	STA ID1,I	
0478	01466	063535R	LDA NAM34	
0479	01467	173312R	STA ID2,I	
0480	01470	063536R	LDA NAM5	
0481	01471	173313R	STA ID3,I	
0482*				
0483	01472	002400	CLA	INSERT COMMON LENGTH
0484	01473	067352R	LDB DISCX	

0485	01474	004010	SLB	
0486	01475	063450R	LDA MSIGN	(BIT 15 SET IF FROM USER DISC)
0487	01476	043541R	ADA NCL	
0488	01477	173314R	STA ID4,I	
0489*				
0490	01500	063543R	LDA DSKNM	DISC ADDRESS OF
0491	01501	173315R	STA ID5,I	NAM RECORD
0492	01502	067537R	LDB NPL	
0493	01503	006021	SSB,RSS	SAVE, IF PROGRAM WAS COMPILED
0494	01504	002400	CLA	
0495	01505	073412R	STA LWH1	
0496*				
0497	01506	063542R	LDA NTYPE	GET PROGRAM TYPE FROM NAM RECORD
0498	01507	010073	AND M177	
0499	01510	067432R	LDB RLGTH	IF BCS RECORD (NAM LENGTH = 9),
0500	01511	054042	CPB N9	FORCE PROGRAM TYPE 3
0501	01512	060056	LDA ,3	
0502	01513	002003	SZA,RSS	IF PROGRAM TYPE = 0,
0503	01514	060056	LDA ,3	FORCE PROGRAM TYPE 3
0504	01515	067537R	LDB NPL	
0505	01516	047541R	ADB NCL	
0506	01517	006003	SZB,RSS	IF TOTAL PROGRAM LENGTH = 0,
0507	01520	060061	LDA ,6	FORCE PROGRAM TYPE 6
0508	01521	173316R	STA ID6,I	SET NEW PROGRAM TYPE IN IDENT
0509*				
0510	01522	002400	CLA	
0511	01523	173317R	STA ID7,I	
0512	01524	073426R	STA MAXC	INITIALIZE MAX RELOC FOR DBL
0513*				
0514	01525	063306R	LDA TIDNT	INCREMENT IDENT TABLE POINTER
0515	01526	073307R	STA PIDNT	
0516*				
0517	01527	067540R	LDB NBPL	ANY BASE PAGE ALLOCATED?
0518	01530	006003	SZB,RSS	
0519	01531	127402R	JMP NAM,I	NO= RETURN
0520	01532	063504R	LDA ERR14	YES= NOT ALLOWED, ERROR 14 &
0521	01533	026631R	JMP ABRTX	ABORT
0522*				
0523	01534	000000	NAM12 NOP	
0524	01535	000000	NAM34 NOP	
0525	01536	000000	NAM5 NOP	
0526	01537	000000	NPL NOP	
0527	01540	000000	NBPL NOP	
0528	01541	000000	NCL NOP	
0529	01542	000000	NTYPE NOP	
0530	01543	000000	DSKNM NOP	


```

0532*
0533* THIS ROUTINE PROCESSES RELOCATABLE DBL RECORDS.
0534* IT IS ASSUMED THAT "FETCH" HAS BEEN CALLED AND
0535* HAS RETURNED A RECORD TYPE 3. IN ADDITION, THE
0536* BUFFER POINTER IS POINTING TO THE 2ND WORD IN
0537* THE RECORD.
0538*
0539* ONE VARIABLE MUST HAVE A PRE-FILLED VALUE:
0540*
0541*     MAXC = MAX RELOCATION ADDRESS IS INITIALLY
0542*           SET TO ZERO IN THE NAM RECORD
0543*           PROCESSOR
0544*
0545* FUNCTION:
0546*
0547*     TO COMPUTE THE LENGTH OF THE CURRENT PROGRAM.
0548*     THIS VALUE IS STORED IN "LWH2" & USED TO
0549*     UPDATE THE NAM RECORD IF THE PROGRAM BEING
0550*     LOADED WAS COMPILED.
0551*
0552 01544 000000 DBL  NOP
0553*
0554 01545 163340R    LDA CURAL,I  SAVE LENGTH OF CURRENT RECORD
0555 01546 010072    AND M77
0556 01547 073410R    STA TEMP1
0557*
0558 01550 016203R    JSB BUMP
0559 01551 016203R    JSB BUMP
0560 01552 167340R    LDB CURAL,I  GET CURRENT RELOCATION ADDRESS
0561 01553 063426R    LDA MAXC     IS IT GREATER THAN CURRENT MAX
0562 01554 003004    CMA,INA       RELOCATION ADDRESS?
0563 01555 040001    ADA B
0564 01556 002020    SSA
0565 01557 127544R    JMP DBL,I    NO- SKIP IT
0566*
0567 01560 063410R    LDA TEMP1     YES- COMPUTE NEW PROGRAM
0568 01561 040001    ADA B          LENGTH
0569 01562 073413R    STA LWH2
0570 01563 077426R    STB MAXC     SET NEW MAX RELOC ADDR
0571 01564 127544R    JMP DBL,I

```

```

0573*
0574* THIS ROUTINE PROCESSES RELOCATABLE ENT & EXT
0575* RECORDS. IT IS ASSUMED THAT "FETCH" HAS BEEN
0576* CALLED AND HAS RETURNED RECORD TYPE 2 OR 4. IN
0577* ADDITION, THE BUFFER POINTER IS POINTING TO THE
0578* SECOND WORD IN THE BUFFER.
0579*
0580* PRE-FILLED VALUES:
0581*
0582*     "A" REGISTER = 0   IF EXT RECORD
0583*                   -1   IF ENT RECORD
0584*
0585* FUNCTIONS:
0586*
0587*     1. CHECKS TO SEE IF ENT/EXT NAMES ARE IN LST
0588*
0589*         A. YES:   ENT: IF LST4>0, ERROR 7 (DUP ENT)
0590*                      IF LST4=0, SET LST4 TO ID1
0591*                      IF LST4<0, ERROR 15 (SEG
0592*                          BEFORE MAIN) IF MAIN;
0593*                          OTHERWISE SET LST4 TO
0594*                          ID1
0595*                      EXT: IF MAIN, CONTINUE PROCESS
0596*                          IF SEGMENT & LST4<=0, CONT.
0597*                          IF SEGMENT & LST4>0, FIND ID
0598*                          ENTRY OF ENT NAME. IF MAIN
0599*                          SET ID1 TO ID1 OF DEFINING
0600*                          PROGRAM.
0601*
0602*         B. NO:   ENT: PUT IN LST, SET LST4 TO ID1
0603*
0604*                      EXT: PUT IN LST, SET LST4 TO 0
0605*                          IF NOT SEGMENT; TO COMPLE-
0606*                          MENT OF ID1 IF SEGMENT
0607*
0608* 01565 000000 EN,XT NOP
0609* 01566 073410R      STA NXFLG      SAVE ENT/EXT FLAG
0610*
0611* 01567 163340R      LDA CURAL,I     SET COUNTER TO # NAMES
0612* 01570 010071      AND M37         IN RECORD
0613* 01571 003004      CMA,INA
0614* 01572 073371R      STA EXCNT
0615*
0616* 01573 016203R      JSB BUMP        BUMP BUFFER POINTER TO START OF
0617* 01574 016203R      JSB BUMP        SYMBOL NAME
0618*
0619* 01575 163340R SAVNM LDA CURAL,I     SAVE SYMBOL NAME IN TBUF
0620* 01576 072210R      STA TBUF
0621* 01577 016203R      JSB BUMP
0622* 01600 163340R      LDA CURAL,I
0623* 01601 072211R      STA TBUF+1
0624* 01602 016203R      JSB BUMP
0625* 01603 163340R      LDA CURAL,I
0626* 01604 010075      AND M1774
0627* 01605 072212R      STA TBUF+2
0628* 01606 016203R      JSB BUMP        POSITION POINTER TO NEXT NAME

```

```

0629*
0630 01607 016230R JSB INLST SEARCH LST FOR NAME
0631 01610 016234R NEXT JSB LSTX
0632 01611 027657R JMP NOENT NOT FOUND
0633 01612 062210R LDA TBUF
0634 01613 153300R CPA LST1,I
0635 01614 002001 RSS
0636 01615 027610R JMP NEXT
0637 01616 062211R LDA TBUF+1
0638 01617 153301R CPA LST2,I
0639 01620 002001 RSS
0640 01621 027610R JMP NEXT
0641 01622 163302R LDA LST3,I
0642 01623 010075 AND M1774
0643 01624 052212R CPA TBUF+2
0644 01625 002001 RSS
0645 01626 027610R JMP NEXT
0646*
0647* NAME FOUND IN LST
0648*
0649 01627 063410R LDA NXFLG ENT OR EXT?
0650 01630 002003 SZA,RSS
0651 01631 027703R JMP EXT1 EXT PROCESSING
0652*
0653 01632 163303R LDA LST4,I ID ADDRESS ALREADY ASSIGNED?
0654 01633 002003 SZA,RSS
0655 01634 027654R JMP SIDA NO- SET ID ADDRESS
0656 01635 002021 SSA,RSS
0657 01636 027646R JMP DUP YES- DUP ENTRY POINT ERROR
0658 01637 163316R LDA ID6,I
0659 01640 010073 AND M177
0660 01641 050056 CPA ,3
0661 01642 002001 RSS
0662 01643 027654R JMP SIDA
0663*
0664 01644 063505R LDA ERR15
0665 01645 026631R JMP ABRTX
0666*
0667 01646 063475R DUP LDA ERR07 DUPLICATE ENTRY POINT ERROR
0668 01647 016344R JSB ERROR
0669 01650 060060 LDA ,5 OUTPUT NAME, THEN ABORT
0670 01651 067300R LDB LST1
0671 01652 016331R JSB SYOUT
0672 01653 026632R JMP ABORT
0673*
0674 01654 063311R SIDA LDA ID1 SET ADDRESS OF ID ENTRY INTO
0675 01655 173303R STA LST4,I LST4
0676 01656 027741R JMP ENXT1
0677*
0678* NAME NOT FOUND IN LST
0679*
0680 01657 062210R NOENT LDA TBUF PUT SYMBOL NAME IN LST
0681 01660 173300R STA LST1,I
0682 01661 062211R LDA TBUF+1
0683 01662 173301R STA LST2,I
0684 01663 062212R LDA TBUF+2

```

0685	01664	173302R	STA LST3,I	
0686	01665	002400	CLA	CLEAR BP LINK IN LST
0687	01666	173304R	STA LST5,I	
0688	01667	063276R	LDA TLST	INCREMENT LST POINTER
0689	01670	073277R	STA PLST	
0690*				
0691	01671	063410R	LDA NXFLG	SET LST4 ACCORDING TO RECORD
0692	01672	002002	SZA	TYPE:
0693	01673	027654R	JMP SIDA	IF ENT, LST4 = ID1
0694	01674	163316R	LDA ID6,I	IF EXT, LST4 = -ID1 IF SEGMENT
0695	01675	010073	AND M177	= 0 IF MAIN
0696	01676	067311R	LDB ID1	
0697	01677	050060	CPA ,5	
0698	01700	007001	CMB,RSS	
0699	01701	006400	CLB	
0700	01702	177303R	STB LST4,I	
0701*				
0702	01703	163316R EXT1	LDA ID6,I	FURTHER PROCESSING NECESSARY IF
0703	01704	010073	AND M177	PROGRAM CONTAINING EXT IS A
0704	01705	050060	CPA ,5	SEGMENT
0705	01706	002001	RSS	
0706	01707	027741R	JMP ENXT1	OTHERWISE, ON TO THE NEXT
0707*				
0708	01710	063311R	LDA ID1	SAVE ID ADDRESS OF SEGMENT
0709	01711	073400R	STA IMAIN	
0710*				
0711	01712	163303R	LDA LST4,I	HAS ERRORESPONDING ENTRY BEEN
0712	01713	002002	SZA	DEFINED?
0713	01714	002020	SSA	
0714	01715	027741R	JMP ENXT1	NO- SKIP
0715	01716	073306R	STA TIDNT	YES- SET TO GET ID'S FOR PROG
0716	01717	016265R	JSB IDX	DEFINING ENT
0717	01720	026637R	JMP RCERR	(SHOULD FIND IT)
0718	01721	163316R	LDA ID6,I	IS PROGRAM CONTAINING ENTRY POIN
0719	01722	002021	SSA,RSS	A SUBROUTINE?
0720	01723	027727R	JMP NTMN	YES- SET FOR NOT MAIN
0721	01724	010073	AND M177	NO - IS IT A MAIN?
0722	01725	050056	CPA ,3	
0723	01726	007401	CCB,RSS	YES, TEMP=-1 FOR MAIN
0724	01727	006400 NTMN	CLB	
0725	01730	077444R	STB TEMP	
0726*				
0727	01731	063400R	LDA IMAIN	RESTORE IDENT ENTRY OF PROGRAM
0728	01732	073306R	STA TIDNT	CONTAINING EXT.
0729*				
0730	01733	016265R	JSB IDX	
0731	01734	026637R	JMP RCERR	
0732*				
0733	01735	037444R	ISZ TEMP	WAS PROGRAM CONTAINING ENT A MAI
0734	01736	027741R	JMP ENXT1	NO- GET NEXT SYMBOL
0735	01737	163303R	LDA LST4,I	YES- SET ID ADDRESS OF MAIN
0736	01740	173317R	STA ID7,I	INTO ID7 OF SEGMENT
0737*				
0738	01741	063410R ENXT1	LDA NXFLG	IF ENT, BUMP POINTER PAST
0739	01742	002002	SZA	RELOCATABLE ADDRESS
0740	01743	016203R	JSB BUMP	

PAGE 0032 #02 ENT/EXT RECORD PROCESSOR

0741	01744	037371R	ISZ EXCNT	DONE WITH ALL SYMBOLS IN RECORD?
0742	01745	027575R	JMP SAVNM	NO= PROCESS NEXT NAME
0743	01746	127565R	JMP EN,XT,I	YES= RETURN

```

0745*
0746* THIS ROUTINE PROCESSES RELOCATABLE END RECORDS.
0747* IT IS ASSUMED THAT "FETCH" HAS BEEN CALLED AND
0748* HAS RETURNED A RECORD TYPE OF 5. IN ADDITION,
0749* THE BUFFER POINTER IS POINTING TO THE 2ND WORD
0750* IN THE RECORD.
0751*
0752* TWO VARIABLES MUST HAVE PRE?ILLED VALUES:
0753*
0754*     LWH1 - "0" IF PROGRAM NOT COMPILED
0755*           "DISC ADDR OF NAM RECORD" IF PROGRAM
0801*           WAS COMPILED
0802*           (SET BY NAM RECORD PROCESSOR)
0803*
0804*     LWH2 - PROGRAM LENGTH
0805*           (SET BY DBL RECORD PROCESSOR)
0806*
0807* FUNCTIONS:
0808*
0809*     1. TO SET BIT 15 OF ID6 FOR THIS PROGRAM
0810*
0811*         "0" - PROGRAM IS A SUBROUTINE (NO LABEL
0812*         ON END STATEMENT)
0813*
0814*         "1" - PROGRAM IS A MAIN/SEGMENT (LABEL
0815*         ON END STATEMENT)
0816*
0817*     2. TO SET THE COMPUTED RECORD LENGTH INTO THE
0818*         NAM RECORD IF THE PROGRAM WAS COMPILED.
0819*         (THIS IS NECESSARY FOR CURRENT PAGE LINK
0820*         PROCESSING.)
0821*
0822 01747 000000 END NOP
0823*
0824 01750 002404 CLA,INA SET BIT 15 OF ID6 IF PROGRAM
0825 01751 113340R AND CURAL,I HAD A LABEL ON THE END
0826 01752 001300 RAR STATEMENT (IE. WAS A MAIN OR
0827 01753 133316R IOR ID6,I A SEGMENT)
0828 01754 173316R STA ID6,I
0829 01755 017757R JSB UPNAM UPDATE NAM RECORD IF NECESSARY
0830*
0831 01756 127747R JMP END,I
0832*
0833* UPDATE NAM RECORD - PROG LENGTH & CHECKSUM
0834*
0835 01757 000000 UPNAM NOP
0836 01760 063412R LDA LWH1 WAS PROGRAM COMPILED?
0837 01761 002003 SZA,RSS
0838 01762 127757R JMP UPNAM,I NO-RETURN
0839*
0840 01763 066515R LDB #WRDS SAVE CURRENT # WORDS FOR
0841 01764 077410R STB TEMP1 DISC TARNFER
0842 01766 067550R LDB ,128 SET TO INPUT 1 SECTOR
0843 01766 076515R STB #WRDS
0844*
0845 01767 067551R LDB ADBUF INPUT NAM RECORD INTO DBUF

```

0046	01770	016465R	JSB DISKI	
0047	01771	063413R	LDA LWH2	SET IN COMPUTED PROGRAM
0048	01772	073615R	STA DBUF+6	LENGTH
0049	01773	063607R	LDA DBUF	COMPUTE NEW CHECKSUM
0050	01774	001727	ALF, ALF	
0051	01775	010073	AND M177	
0052	01776	003004	CMA, INA	
0053	01777	040056	ADA .3	
0054	02000	073454R	STA WDCNT	
0055	02001	063610R	LDA DBUF+1	
0056	02002	067551R	LDB ADBUF	
0057	02003	044056	ADB .3	
0058	02004	140001	ADA B, I	
0059	02005	006004	INB	
0060	02006	037454R	ISZ WDCNT	
0061	02007	026004R	JMP +-3	
0062	02010	073611R	STA DBUF+2	
0063*				
0064	02011	063412R	LDA LWH1	OUTPUT UPDATED NAM RECORD
0065	02012	067551R	LDB ADBUF	BACK TO DISC
0066	02013	016473R	JSB DISKO	
0067*				
0068	02014	063410R	LDA TEMP1	RESTOR # WORDS
0069	02015	072515R	STA #WRDS	FOR DISC TRANSFER
0070*				
0071	02016	127757R	JMP UPNAM, I	

```

0073*
0074* SETUP - SETS UP THE RETURN ADDRESS FOR BLOCK
0075* ROUTINE WHEN ALL SECTORS HAVE BEEN INPUT
0076* SETS UP THE BLOCK SIZE (BOTH # SECTORS &
0077* # WRDS) FOR DISC INPUT, ACCORDING TO
0078* TDIR.
0079*
0080* DEPENDENT VARIABLES:
0081*
0082* "A" REGISTER - RETURN ADDRESS WHEN ALL
0083* SECTORS HAVE BEEN INPUT
0084*
0085* TDIR - IF JBIN (=1), BLOCK IS 1
0086* - IF NOT JBIN (=0), BLOCK
0087* IS 4 SECTORS
0088*
0089 02017 000000 SETUP NOP
0090 02020 072103R STA RTRN SET RETURN FOLLOWING COMPLETION
0091*
0092 02021 002404 CLA,INA IF INPUT IS FROM JOB BINARY AREA
0093 02022 067443R LDB TDIR BLOCK SIZE IS 1 SECTOR.
0094 02023 006003 SZB,RSS OTHERWISE, BLOCK SIZE IS 4
0095 02024 040056 ADA ,3 SECTORS.
0096 02025 072105R STA BLCK#
0097 02026 016003X MPY ,128 ALTER # WORDS PER DISC INPUT
0098 02030 072515R STA #WRDS
0099*
0100 02031 126017R JMP SETUP,I

```

```

0102*
0103* BLOCK - INPUTS A BLOCK OF RELOCATABLE RECORDS
0104* FROM THE DISC INTO CORE AT "LBUF". THE
0105* SIZE OF THE BLOCK IS DETERMINED BY
0106* "SETUP". IF THE ENTIRE BLOCK CANNOT BE
0107* INPUT, AS MUCH AS POSSIBLE IS. IF NONE
0108* IS POSSIBLE, RETURN IS MADE THROUGH
0109* "RTRN" (SET BY "SETUP"), AFTER RESTORING
0110* THE NORMAL DISC INPUT TRANSFER TO 1
0111* SECTOR (128 WORDS).
0112* FOLLOWING EACH TRANSFER, THE BUFFER
0113* POINTER & LENGTH ("CURL" & "LCNT") ARE
0114* INITIALIZED; # SECTORS REMAINING IS
0115* DECREMENTED; DISC ADDRESS & # SECTORS
0116* INPUT ARE INCREMENTED.
0117*

```

```

0118* DEPENDENT VARIABLES:
0119*
0120* RTRN - RETURN ADDRESS FOLLOWING COMPLETION
0121* (SET BY SETUP)
0122*
0123* DISCX= "2" FOR SYSTEM DISC
0124* "3" FOR USER DISC
0125*
0126* TDIR = "0" FOR NON-JBIN FILES

```



```

0127*          "1" FOR JBIN FILES (BACKWARD INCR)
0128*
0129*          COUNT= # SECTORS TO BE INPUT
0130*
0131*          BLCK#= BLOCK SIZE OF INPUT TRANSFER
0132*
0133*          BCKAD= STARTING DISC ADDRESS
0134*
0135 02032 000000 BLOCK NOP
0136*
0137 02033 062104R      LDA BLKAD      SAVE CURRENT BLOCK
0138 02034 072106R      STA CBLA      STARTING ADDRESS
0139*
0140 02035 066105R      LDB BLCK#     CAN INPUT DESIRED BLOCK SIZE?
0141 02036 007004      CMB,INB
0142 02037 047335R      ADB COUNT
0143 02040 006020      SSB
0144 02041 026067R      JMP BLCK1     NO= SEE IF ANY INPUT POSSIBLE
0145*
0146 02042 077335R BLCK2 STB COUNT     UPDATE # SECTORS LEFT TO INPUT
0147 02043 062104R      LDA BLKAD
0148 02044 067553R      LDB ALBUF     INPUT BLOCK FROM DISC ADDR IN
0149 02045 016465R      JSB DISKI     BLKAD INTO LBUF
0150*
0151 02046 063553R      LDA ALBUF     INITIALIZE BUFFER POINTER
0152 02047 073340R      STA CURAL
0153 02050 062105R      LDA BLCK#     INITIALIZE BUFFER LENGTH
0154 02051 016003X      MPY ,128     ( = # WORDS)
0155 02053 003004      CMA,INA
0156 02054 073403R      STA LCNT
0157*
0158 02055 062105R      LDA BLCK#
0159 02056 003004      CMA,INA
0160 02057 073444R      STA TEMP
0161 02060 037367R BLCK5 ISZ DSCNT     INCREMENT # SECTORS INPUT
0162 02061 062104R      LDA BLKAD     INCREMENT DISC ADDRESS
0163 02062 017443R      JSB NDA
0164 02063 072104R BLCK3 STA BLKAD
0165 02064 037444R      ISZ TEMP
0166 02065 026060R      JMP BLCK5
0167 02066 126032R      JMP BLOCK,I
0168*
0169 02067 063335R BLCK1 LDA COUNT     ANY MORE SECTORS TO INPUT?
0170 02070 002003      SZA,RSS
0171 02071 026100R      JMP BLCK6     NO= RETURN
0172 02072 072105R      STA BLCK#     YES= SET BLCK# TO # SECTORS
0173 02073 016003X      MPY ,128
0174 02075 072515R      STA #WRDS
0175 02076 006400      CLB           LEFT; SET # SECTORS LEFT
0176 02077 026042R      JMP BLCK2     TO ZERO.
0177*
0178 02100 063550R BLCK6 LDA ,128     RESTORE # WORDS PER DISC INPUT
0179 02101 072515R      STA #WRDS     TRANSFER TO 128
0180 02102 126103R      JMP RTRN,I    RETURN
0181*
0182 02103 000000 RTRN  NOP           RETURN ADDRESS

```

PAGE 0037 #03 UTILITIES FOR RELOCATABLE DISC INPUT(PASS 1)

0183	02104	000000	BLKAD	NOP	STARTING DISC ADDRESS
0184	02105	000000	BLCK#	NOP	BLOCK SIZE
0185	02106	000000	CBLA	NOP	CURRENT BLCK DISC ADDRESS

0187*
 0188* NOTE: FETCH, BUMP, & SKIP ROUTINES ARE ALSO
 0189* USED WHEN PROCESSING RELOCATABLE INPUT
 0190* FROM PERIPHERAL DEVICES. IN THESE CASES,
 0191* IT IS ASSUMED THAT NO RECORD LENGTH WILL
 0192* EXCEED 128 WORDS.

0194*
 0195* FETCH - SETS UP THE RECORD LENGTH OF THE CURRENT
 0196* RELOCATABLE RECORD (RLGTH & SCNT)
 0197* VERIFIES RECORD VALIDITY:
 0198* IF ILLEGAL LENGTH OR TYPE = ERROR 2
 0199*
 0200* NOTE: IF A ZERO RECORD LENGTH IS EN-
 0201* COUNTERED, THE RECORD IS ASSUMED TO
 0202* CONTAIN THE ZEROES FOLLOWING THE END
 0203* RECORD. THE REST OF THE SECTOR IS
 0204* SKIPPED

0205*
 0206* DEPENDENT VARIABLES:

0207*
 0208* CURAL - ASSUMED TO BE POINTING TO 1ST
 0209* WORD IN RELOCATABLE RECORD ON
 0210* ENTRY. ON EXIT, IT IS POINTING
 0211* TO THE 2ND WORD.
 0212*

0213 02107 000000 FETCH NOP

0214*
 0215 02110 163340R LDA CURAL,I
 0216 02111 002020 SSA
 0217 02112 026637R JMP RCERR ILLEGAL RECORD LENGTH
 0218 02113 001727 ALF,ALF
 0219 02114 010073 AND M177
 0220 02115 003004 CMA,INA
 0221 02116 073432R STA RLGTH SET VALID RECORD LENGTH
 0222*
 0223 02117 002003 SZR,RSS RECORD LENGTH=0?
 0224 02120 026171R JMP NEWS YES- GET NEXT SECTOR
 0225 02121 072202R STA SCNT NO- SET RECORD LENGTH FOR SKIP
 0226*
 0227 02122 040056 ADA ,3 VALID RECORD LENGTH IS
 0228 02123 002021 SSA,RSS 4 - 60
 0229 02124 026637R JMP RCERR
 0230 02125 073454R STA WDCNT
 0231 02126 043546R ADA D57
 0232 02127 002020 SSA
 0233 02130 026637R JMP RCERR
 0234*
 0235 02131 016203R JSB BUMP BUMP POINTER TO 2ND WORD
 0236*
 0237 02132 163340R CHKOK LDA CURAL,I GET RECORD TYPE.
 0238 02133 001723 ALF,RAR VALID IF 1-5
 0239 02134 010062 AND ,7

0240	02135	070001	STA B	
0241	02136	002002	SZA	
0242	02137	044045	ADB N6	
0243	02140	006021	SSB, RSS	
0244	02141	026637R	JMP RCERR	NOT VALID - ERROR 2
0245*				
0246	02142	050054	CPA .1	NAM RECORD?
0247	02143	002001	RSS	YES- UPDATE DISC ADDR
0248	02144	126107R	JMP FETCH, I	CONTAINS RECORD TYPE
0249*				
0250	02145	007400	CCB	COMPUTE NEW DISC ADDRESS
0251	02146	063403R	LDA LCNT	FOR NAM RECORD
0252	02147	040052	ADA N1	
0253	02150	016002X	DIV N128	
0254	02152	066105R	LDB BLCK#	
0255	02153	007004	CMB, INB	
0256	02154	040001	ADA B	
0257	02155	002021	SSA, RSS	
0258	02156	026167R	JMP NEW2	
0259	02157	073442R	STA TCNT	
0260*				
0261	02160	062106R	LDA CBLA	
0262	02161	017443R	JSB NDA	
0263	02162	037442R	ISZ TCNT	
0264	02163	026161R	JMP *-2	
0265	02164	073543R	STA DSKNM	
0266	02165	002404	CLA, INA	"A" = NAM RECORD TYPE
0267	02166	126107R	JMP FETCH, I	
0268*				
0269	02167	062106R	LDA CBLA	
0270	02170	026164R	JMP NEW1	
0271*				
0272	02171	007400	CCB	POSITION POINTERS TO
0273	02172	063403R	LDA LCNT	BEGINNING OF NEXT
0274	02173	016002X	DIV .128	SECTOR
0275	02175	006003	SZB, RSS	
0276	02176	067547R	LDB N128	
0277	02177	076202R	STB SCNT	
0278	02200	016215R	JSB SKIP	
0279	02201	026110R	JMP FETCH+1	REPROCESS NAM RECORD
0280	02202	000000	SCNT NOP	

0282*
 0283* BUMP = INCREMENTS BUFFER ADDRESS POINTER, 8
 0284* DECREMENTS # WORDS WITHIN RELOCATABLE
 0285* RECORD,
 0286* DECREMENTS # WORDS IN BUFFER
 0287* WHEN ALL WORDS HAVE BEEN PROCESSED, "BLOCK"
 0288* IS CALLED TO READ IN NEXT BLOCK OF
 0289* SECTORS
 0290*
 0291* DEPENDENT VARIABLES:
 0292*
 0293* CURAL = CURRENT BUFFER ADDRESS POINTER

```

0294*
0295*      LCNT  = # WORDS REMAINING IN BUFFER
0296*
0297*      SCNT  = # WORDS REMAINING IN RECORD
0298*
0299 02203 000000 BUMP  NOP
0300*
0301 02204 037340R      ISZ CURAL      INCREMENT ADDR POINTER
0302 02205 036510R      ISZ W,PRG      ICREMENT # WORDS IN PROGRAM
0303 02206 000000      NOP              (IGNORE ZERO)
0304 02207 036202R      ISZ SCNT       INCREMENT # WORDS IN RECORD
0305 02210 000000      NOP              (IGNORE ZERO)
0306 02211 037403R      ISZ LCNT       INCREMENT # WORDS IN BUFFER
0307 02212 126203R      JMP BUMP,I     NEW BLOCK NOT NEEDED
0308*
0309 02213 016032R      JSB BLOCK      INPUT NEW BLOCK
0310 02214 126203R      JMP BUMP,I

0312*
0313* SKIP = SKIPS TO THE BEGINNING OF THE NEXT
0314*      RELOCATABLE RECORD IN THE INPUT BUFFER
0315*      (DOES NOT SKIP IF ALREADY POSITIONED TO
0316*      BEGINNING OF RECORD = IE SCNT=0)
0317*
0318*      DEPENDENT VARIABLES:
0319*
0320*      SCNT  = # WORDS REMAINING IN RECORD
0321*      (SET BY "FETCH", UPDATED BY "BUMP")
0322*
0323 02215 000000 SKIP  NOP
0324*
0325 02216 062202R      LDA SCNT        ALREADY AT BEGINNING OF RECORD?
0326 02217 002021      SSA,RSS
0327 02220 126215R      JMP SKIP,I     YES= RETURN
0328*
0329 02221 073442R      STA TCNT        NO = BUMP POINTERS PAST END
0330 02222 016203R      JSB BUMP        OF CURRENT RECORD
0331 02223 037442R      ISZ TCNT
0332 02224 026222R      JMP *-2
0333 02225 126215R      JMP SKIP,I

```

0335*

0336* VERIFY THAT NO DUPLICATE FILE NAMES EXIST

0337*

```

0338 02226 002400 LMAIN CLA
0339 02227 073425R STA PTYPE
0340 02230 016261R JSB INIDX CLEAR PROGRAM USEAGE FLAGS
0341 02231 016265R CLRID JSB IDX SET CURRENT IDENT ADDRESSES
0342 02232 026277R JMP IDCLR
0343 02233 163313R LDA ID3,I
0344 02234 010075 AND M1774
0345 02235 173313R STA ID3,I CLEAR USEAGE FLAG
0346 02236 163316R LDA ID6,I CHECK PROGRAM TYPE
0347 02237 010073 AND M177
0348 02240 050056 CPA ,3 CHECK IF MAIN
0349 02241 026254R JMP USCHK
0350 02242 050057 CPA ,4 CHECK IF DEVICE DRIVER
0351 02243 026275R JMP DDRV
0352 02244 050060 CPA ,5 CHECK IF SEGMENT
0353 02245 026254R JMP USCHK
0354 02246 002002 SZA CHECK IF TYPE 0
0355 02247 026231R JMP CLRID NO, SO NOT MAIN
0356 02250 163316R CLR0 LDA ID6,I CONVERT TYPES 0 & 4 TO 3
0357 02251 010075 AND M1774
0358 02252 030056 IOR ,3
0359 02253 173316R STA ID6,I
0360 02254 063311R USCHK LDA ID1
0361 02255 072261R STA ++4
0362 02256 016001X JSB EXEC
0363 02257 002263R DEF ++4
0364 02260 007436R DEF ,18 REQUEST CODE
0365 02261 002261R DEF * FILE NAME ADDRESS
0366 02262 005441R DEF SECT * SECTORS
0367 02263 063441R LDA SECT GET # SECTORS IN FILE
0368 02264 002003 SZA,RSS NON-ZERO--ALREADY IN FILE
0369 02265 026231R JMP CLRID TRY NEXT PROGRAM
0370 02266 063474R LDA ERR06 PRINT DUPL FILE NAME ERROR CODE
0371 02267 016344R JSB ERROR
0372 02270 060060 LDA ,5 PRINT FILE NAME
0373 02271 067311R LDB ID1
0374 02272 016331R JSB SYOUT
0375 02273 037425R ISZ PTYPE INCR DUPL FILE FLAG
0376 02274 026231R JMP CLRID SEE IF ANY MORE DUPL FILES
0377*
0378 02275 073462R DDRV STA DVR SET DRIVER FLAG # 0
0379 02276 026250R JMP CLR0 SET TYPE TO 3 TEMPORARILY
0380 02277 063425R IDCLR LDA PTYPE CHECK IF ANY DUPL FILE FLAG SET
0381 02300 002002 SZA NO
0382 02301 026632R JMP ABORT YES-SO ABORT LOADER

```

0383*

0384* INITIALIZE DIRECTORY

0385*

```

0386 02302 060200 LDA DISCL DISCL+1 = START T/S OF USER DIR
0387 02303 002004 INA
0388 02304 073357R STA DSKID
0389 02305 063552R DRSCH LDA AIBUF SET INITIAL IBUF POINTER
0390 02306 073337R STA CURAI

```

0391	02307	063547R	LDA N128	SET IBUF COUNT
0392	02310	073375R	STA ICNT	
0393	02311	016000R	JSB SETCR	SET USER DISC FLAG IN BASE PAGE
0394	02312	063357R	LDA DSKID	A IS DISC ADDRESS
0395	02313	067552R	LDB AIBUF	B IS CORE ADDRESS
0396	02314	016465R	JSB DISKI	READ IN 1 SECTOR OF USER DIR
0397	02315	163337R	DRCHK LDA CURAI,1	
0398	02316	002003	SZA,RSS	CHECK IF AT END OF DIRECTORY
0399	02317	026357R	JMP ENDDR	END OF USER DIR FOUND
0400	02320	050052	CPA N1	CHECK IF AT END OF DIR SECTOR
0401	02321	002001	RSS	END OF SECTOR
0402	02322	026332R	JMP DRCH1	NOT AT END OF SECTOR
0403	02323	060174	DRCH0 LDA TSONE	INCREMENT SECTOR ADDRESS
0404	02324	073357R	STA DSKID	
0405	02325	010074	AND M377	
0406	02326	002002	SZA	
0407	02327	026305R	JMP DRSCH	GO READ IN NEXT SECTOR
0408	02330	063500R	DRCHA LDA ERR10	
0409	02331	026631R	JMP ABRTX	OUT OF DIRECTORY SPACE
0410	02332	067337R	DRCH1 LDB CURAI	POINTER IN IBUF
0411	02333	044055	ADB .2	INCR TO NAME 5 - TYPE ADDRESS
0412	02334	160001	LDA B,1	
0413	02335	010073	AND M177	ISOLATE TYPE
0414	02336	040045	ADA N6	CHECK IF 5 OR 11 WORD ENTRY
0415	02337	002021	SSA,RSS	SKIP=11 WORD DIRECTORY ENTRY
0416	02340	026347R	JMP DRCH5	5 WORD DIRECTORY ENTRY
0417	02341	044064	ADB .9	PROCESS 11 WORD ENTRY
0418	02342	077337R	STB CURAI	
0419	02343	067375R	LDB ICNT	
0420	02344	047432R	ADB .11	
0421	02345	077375R	STB ICNT	
0422	02346	026354R	JMP DRCH6	GO TRY NEXT ENTRY
0423	02347	044056	DRCH5 ADB .3	PROCESS 5 WORD ENTRY
0424	02350	077337R	STB CURAI	
0425	02351	067375R	LDB ICNT	
0426	02352	044060	ADB .5	
0427	02353	077375R	STB ICNT	
0428	02354	006021	DRCH6 SSB,RSS	CHECK IF PAST END OF SECTOR
0429	02355	026323R	JMP DRCH0	
0430	02356	026315R	JMP DRCHK	TRY NEXT ENTRY
0431	02357	063375R	ENDDR LDA ICNT	CHECK IF ROOM FOR MORE ENTRIES
0432	02360	040065	ADA .10	
0433	02361	002020	SSA	NO=SKIP
0434	02362	026400R	JMP LDABS	YES
0435	02363	003400	CCA	
0436	02364	017662R	JSB OUTID	OUTPUT MORE DIRECTORY FOLLOWS FLA
0437	02365	067375R	LDB ICNT	
0438	02366	057547R	CPB N128	CHECK IF NEW SECTOR
0439	02367	026375R	JMP DRCLR	GO CLEAR NEXT DRCTRY SECTOR
0440	02370	077444R	STB TEMP	SAVE ZERO OUTPUT COUNT
0441	02371	002400	CLA	
0442	02372	017662R	JSB OUTID	OUTPUT ZERO TO DIRECT, BUFFER
0443	02373	037444R	ISZ TEMP	
0444	02374	026371R	JMP *-3	GO OUTPUT ANOTHER ZERO
0445	02375	063357R	DRCLR LDA DSKID	
0446	02376	067552R	LDB AIBUF	

```

0447 02377 016473R JSB DISKO CLEAR NEXT DRCTRY SECTOR
0448*
0449* LOAD RELOCATABLES & CREATE ABSOLUTE CODE
0450*
0451 02400 063460R LDABS LDA UMFNA SET INITIAL PROG RELOC ADDRESS
0452 02401 073330R STA BGBND
0453 02402 073605R STA COMAD
0454 02403 073427R STA RELAD
0455 02404 073423R STA PPREL
0456 02405 063463R LDA LISTU
0457 02406 010072 AND M77
0458 02407 033443R IOR M1100
0459 02410 073444R STA TEMP
0460 02411 016001X JSB EXEC CALL EXEC TO PAGE LIST DEVICE
0461 02412 002416R DEF ++4
0462 02413 000056 DEF ,3
0463 02414 011444R DEF TEMP
0464 02415 000041 DEF N10
0465 02416 063436R PRHED LDA ,18 PRINT "RELOCATING LOADER" HEADING
0466 02417 067501R LDB DOSHD
0467 02420 016364R JSB DRKEY
0468 02421 016355R JSB SPACE
0469 02422 063437R LDA ,36 PRINT "NAME/BOUNDS" HEADING
0470 02423 067513R LDB NAENT
0471 02424 016364R JSB DRKEY
0472 02425 016355R JSB SPACE
0473 02426 063547R LDA N128
0474 02427 073350R STA DCNT INITIALIZE ABSOLUTE OUTPUT COUNT
0475 02430 067551R LDB ADBUF
0476 02431 077336R STB CURAD SET CURRENT DBUF ADDRESS
0477 02432 063353R LDA DSKAB
0478 02433 073354R STA DSKAD SET CURRENT ABS DISC ADDRESS
0479 02434 060056 LDA ,3 SET PROGRAM TYPE TO USER MAIN
0480 02435 073425R STA PTYPE
0481 02436 063423R LDA PPREL
0482 02437 073455R STA XPREL
0483 02440 063305R LDA BIDNT
0484 02441 073310R STA CIDNT SET CIDNT FOR IDSCN
0485 02442 063564R LDA PLAD
0486 02443 073341R STA CURAP
0487 02444 002400 CLA
0488 02445 073603R STA LIBRY
0489 02446 073415R STA PGCNT
0490 02447 037404R ISZ LDING SET LOADING FLAG
0491 02450 016451R BDLD JSB IDSCN SCAN FOR USER MAIN
0492 02451 026654R JMP BDEND END OF USER MAIN
0493 02452 167313R LDB ID3,I GET USAGE FLAG
0494 02453 006014 SLB,INB SKIP=PROG NOT LOADED
0495 02454 026450R JMP BDLD PROG LOADED
0496 02455 177313R STB ID3,I SET PROGRAM LOADED FLAG
0497 02456 163314R LDA ID4,I WD 4 OF IDENT TABLE FOR COMMON
0498 02457 001665 ELA,CLE,ERA REMOVE BIT 15
0499 02460 043455R ADA XPREL ADD COMMON LENGTH TO BASE
0500 02461 073423R STA PPREL SET NEW RELOC ADDRESS
0501 02462 063311R LDA ID1 SAVE MAIN IDENT ADDRESS FOR
0502 02463 073376R STA IDMB5 SEGMENT REFERENCE

```


0503	02464	016123R	JSB DINIT	INITIALIZE DISC STORE ADDRESSES
0504	02465	016355R	JSB SPACE	
0505	02466	016774R	JSB LOAD	LOAD USER MAIN PROGRAM
0506	02467	063603R	LDA TPREL	SAVE END OF MAIN AS LAST
0507	02470	072771R	STA MAXSG	ADDRESS OF LARGEST MODULE
0508	02471	063376R	LDA ICNT	SAVE POINTERS NECESSARY TO
0509	02472	072772R	STA SAVE1	DEFINE DISC ADDRESS AND
0510	02473	063357R	LDA DSKID	OFFSET WITHIN SECTOR OF
0511	02474	072773R	STA SAVE2	MAIN PROGRAM'S DIRECTORY
0512	02475	017461R	JSB GENID	GENERATE DIRECTORY ENTRY
0513	02476	016115R	JSB REMDO	OUTPUT REMAINDER OF RECORD
0514	02477	037415R	ISZ PGCNT	
0515	02500	016133R	JSB RESET	GO RESET DSKAD TO PROPER ADDRESS
0516	02501	002400	CLA	
0517	02502	073603R	STA LIBRY	
0518	02503	063354R	LDA DSKAD	
0519	02504	073356R	STA DSKBS	
0520	02505	063441R	LDA TBREL	
0521	02506	067413R	LDB PBREL	
0522	02507	016341R	JSB BPOUT	OUTPUT BP SECTION
0523	02510	063603R	LDA TPREL	
0524	02511	073325R	STA BSPAD	SAVE PROG RELOC ADDR FOR BS
0525	02512	063441R	LDA TBREL	
0526	02513	073324R	STA BSBAD	SET BP RELOC ADDR FOR BS
0527	02514	060060	LDA ,5	SET CURRENT TYPE TO SSER SEGMENT
0528	02515	073425R	STA PTYPE	
0529	02516	063341R	LDA CURAP	
0530	02517	002004	INA	
0531	02520	073422R	STA PLMAN	SAVE CURRENT PROGRAM LIST ADDR
0532	02521	016261R	JSB INIDX	
0533	02522	016265R	JSB IDX	SET IDENT ADDRESSES
0534	02523	026602R	JMP BSEND	
0535	02524	063311R	LDA ID1	
0536	02525	073400R	STA IMAIN	SAVE MAIN BS IDENT ADDRESS
0537	02526	163316R	LDA ID6,I	
0538	02527	002021	SSA,RSS	
0539	02530	026522R	JMP BSLD	
0540	02531	010073	AND M177	
0541	02532	050060	CPA ,5	SEGMENT?
0542	02533	002001	RSS	YES
0543	02534	026522R	JMP BSLD	NO- TRY NEXT ENTRY
0544	02535	163317R	LDA ID7,I	GET BS MAIN IDENT ADDRESS
0545	02536	002020	SSA	
0546	02537	003004	CMA,INA	
0547	02540	053376R	CPA IDMB5	DOES THIS CALL USER MAIN
0548	02541	002001	RSS	YES
0549	02542	026522R	JMP BSLD	NO-TRY NEXT ENTRY
0550	02543	063306R	LDA TIDNT	SAVE ADDR FOR NEXT SEGMENT SCAN
0551	02544	073320R	STA ABSID	
0552	02545	007400	CCB	
0553	02546	077374R	STB HDFLG	SET HEADING FOR SEGMENT
0554	02547	016123R	JSB DINIT	
0555	02550	016355R	JSB SPACE	
0556	02551	017005R	JSB LOADS	LOAD USER SEGMENT
0557	02552	017461R	JSB GENID	GENERATE DIRECTORY ENTRY
0558	02553	016115R	JSB REMDO	OUTPUT REMAINDER OF ABS RECORD

0559	02554	016133R		JSB RESET	
0560	02555	002400		CLA	
0561	02556	073603R		STA LIBRY	
0562	02557	063441R		LDA TBREL	
0563	02560	067324R		LDB BSBAD	
0564	02561	016341R		JSB BPOUT	OUTPUT BASE PAGE SECTION
0565	02562	063324R		LDA BSBAD	
0566	02563	067441R		LDB TBREL	
0567	02564	016277R		JSB CLRLT	CLEAR BP LINKAGES
0568	02565	067603R		LDB TPREL	KEEP TRACK OF LARGEST SEQUENT
0569	02566	062771R		LDA MAXSG	
0570	02567	003004		CMA, INA	
0571	02570	043603R		ADA TPREL	
0572	02571	002021		SSA, RSS	
0573	02572	076771R		STB MAXSG	
0574	02573	063324R		LDA BSBAD	
0575	02574	073441R		STA TBREL	SET BP RELOC ADDR
0576	02575	063325R		LDA BSPAD	
0577	02576	073603R		STA TPREL	SET PROG RELOC ADDR
0578	02577	063320R		LDA ABSID	
0579	02600	073306R		STA TIDNT	
0580	02601	026622R		JMP BSLD	GO LOAD NEXT SEGMENT
0581	02602	063354R	BSEND	LDA DSKAD	
0582	02603	073355R		STA DSKBR	SAVE CURRENT DISC ADDR OF ABS
0583	02604	063356R		LDA DSKBS	
0584	02605	073354R		STA DSKAD	SET CURRENT BP CODE ADDRESS
0585	02606	063324R		LDA BSBAD	
0586	02607	067413R		LDB PBREL	
0587	02610	016341R		JSB BPOUT	OUTPUT BP CODE FOR MAIN
0588	02611	062771R		LDA MAXSG	HAS A SEGMENT BEEN LOADED?
0589	02612	053325R		CPA BSPAD	
0590	02613	026641R		JMP BS1	NO= SKIP DIRECTORY UPDATE
0591	02614	062773R		LDA SAVE2	HAS SECTOR CONTAINING MAIN'S
0592	02615	053357R		CPA DSKID	DIRECTORY ENTRY BEEN OUTPUT?
0593	02616	026633R		JMP INCOR	NO= UPDATE IN CORE ONLY
0594	02617	067551R		LDB ADBUF	YES= DISC UPDATE REQUIRED
0595	02620	016465R		JSB DISKI	
0596	02621	066772R		LDB SAVE1	COMPUTE NECESSARY OFFSET INTO
0597	02622	047550R		ADB ,128	MAIN'S DIRECTORY ENTRY
0598	02623	047551R		ADB ADBUF	
0599	02624	044065		ADB ,10	
0600	02625	062771R		LDA MAXSG	STORE LARGEST ADDRESS IN WORD 11
0601	02626	170001		STA B,I	OF MAIN'S DIRECTORY ENTRY
0602	02627	062773R		LDA SAVE2	RE-WRITE DIRECTORY ENTRY
0603	02630	067551R		LDB ADBUF	
0604	02631	016473R		JSB DISKO	
0605	02632	026641R		JMP BS1	
0606	02633	066772R	INCOR	LDB SAVE1	COMPUTE OFFSET IN MEMORY BUFFER
0607	02634	047550R		ADB ,128	
0608	02635	047552R		ADB AIBUF	
0609	02636	044065		ADB ,10	
0610	02637	062771R		LDA MAXSG	STORE LARGEST SEGMENT VALUE IN
0611	02640	170001		STA B,I	MAIN'S DIRECTORY ENTRY
0612	02641	063355R	BS1	LDA DSKBR	
0613	02642	073354R		STA DSKAD	SET CURRENT ABS DISC ADDRESS
0614	02643	063413R		LDA PBREL	

0615	02644	067324R	LDB	BSBAD	
0616	02645	016277R	JSB	CLRLT	CLEAR BP LINKAGES
0617	02646	060056	LDA	,3	
0618	02647	073425R	STA	PTYPE	SET PROG TYPE TO USER MAIN
0619	02650	016527R	JSB	CLID3	CLEAR PROG LOADED FLAG
0620	02651	063564R	LDA	PLAD	
0621	02652	073341R	STA	CURAP	RESET PLIST POINTER
0622	02653	026450R	JMP	BDLD	LOAD NEXT USER MAIN
0623	02654	016355R	BDEND	JSB	SPACE
0624	02655	063415R	LDA	PGCNT	CHECK IF ANY PROGRAMS LOADED
0625	02656	002002	SZA		SKIP=NO PROGRAMS LOADED
0626	02657	026664R	JMP	BD005	
0627	02660	063436R	LDA	,18	
0628	02661	066757R	LDB	NOLOD	
0629	02662	016331R	JSB	SYOUT	
0630	02663	026667R	JMP	BD010	
0631	02664	063434R	BD005	LDA	,14
0632	02665	066747R	LDB	CMESS	
0633	02666	016331R	JSB	SYOUT	
0634	02667	016636R	BD010	JSB	SET
0635*					
0636*	LOADER TERMINATION				
0637*					
0638	02670	002400	LTERM	CLA	
0639	02671	017662R	JSB	OUTID	
0640	02672	016000R	JSB	SETCR	SET USER DISC FLAG IN BASE PAGE
0641	02673	063357R	LDA	DSKID	
0642	02674	002003	SZA	,RSS	
0643	02675	026725R	JMP	LTERZ	
0644	02676	067552R	LDB	AIBUF	
0645	02677	016473R	JSB	DISKO	OUTPUT LAST DIRECTORY BUFFER
0646	02700	063415R	LDA	PGCNT	
0647	02701	002003	SZA	,RSS	
0648	02702	026725R	JMP	LTERZ	
0649	02703	063354R	LDA	DSKAD	UPDATE USER FILE T/S BOUNDARY
0650	02704	067561R	LDB	DFUTS	IN BASE PAGE
0651	02705	016004R	JSB	BPSTR	
0652	02706	060155	LDA	SYSSC	
0653	02707	050161	CPA	CUDSC	
0654	02710	002001	RSS		
0655	02711	026715R	JMP	LTERX	OK = USER NOT ON SYSTEM
0656	02712	063354R	LDA	DSKAD	
0657	02713	067562R	LDB	DFSTS	LIKEWISE SET SYNTS = UDNTS
0658	02714	016004R	JSB	BPSTR	STORE A IN B
0659	02715	060200	LTERX	LDA	DISCL
0660	02716	067551R	LDB	ADBUF	
0661	02717	016465R	JSB	DISKI	
0662	02720	060157	LDA	UDNTS	NEXT T/S AVAILABLE IN USER AREA
0663	02721	073707R	STA	DBUF+64	
0664	02722	060200	LDA	DISCL	
0665	02723	067551R	LDB	ADBUF	WRITE OUT SYS BUFFER SECTOR
0666	02724	016473R	JSB	DISKO	
0667	02725	060331	LTERZ	LDA	SSFLG
0668	02726	053507R	CPA	ASCNO	RESTORE SS FLAG
0669	02727	026733R	JMP	LTERW	
0670	02730	063433R	LDA	SAVSS	

PAGE 0047 #03 DOS-III LOADER - (PASS 2) LOAD RELOCATABLES

0671	02731	067510R	LDB	XSSFL	
0672	02732	016004R	JSB	BPSTR	RESTORE SS FLAG IN BASE PAGE
0673	02733	016001X	LTERW	JSB	EXEC REQUEST PROGRAM COMPLETION
0674	02734	002736R	DEF	++2	
0675	02735	000061	DEF	,6	
0676*					
0677	02736	002737R	MESAB	DEF	++1
0678	02737	046117	ASC	8,LOADR	TERMINATED
0679	02747	002750R	CMESS	DEF	++1
0680	02750	046117	ASC	7,LOADR	COMPLETE
0681	02757	002760R	NOLOD	DEF	++1
0682	02760	047117	ASC	9,NO	PROGRAMS LOADED
0683	02771	000000	MAXSG	NOP	LARGEST SEGMENT ADDRESS
0684	02772	000000	SAVE1	NOP	SAVED DIRECTORY OFFSET
0685	02773	000000	SAVE2	NOP	SAVED DIRECTORY DISC ADDRESS

0687* LOAD PROVIDES ENTRY TO LOADS FOR THOSE PROGRAMS WHICH REQUIRE
 0688* USE OF A NEW BASE PAGE AND PROGRAM BASE.
 0689*

0690	02774	000000	LOAD	NOP	
0691	02775	007400		CCB	SET HEADING FLAG
0692	02776	077374R		STB H0FLG	
0693	02777	063423R		LDA PPREL	
0694	03000	073603R		STA TPREL	SET CURRENT PROG RELOC ADDRESS
0695	03001	063413R		LDA PBREL	
0696	03002	073441R		STA TBREL	SET CURRENT BP RELOC ADDRESS
0697	03003	017005R		JSB LOADS	
0698	03004	126774R		JMP LOAD,I	RETURN
0699*					

0700* LOADS IS THE MAIN LOADING ROUTINE FOR GENERATING THE ABSOLUTE
 0701* CODE AND LINKING ALL CALLED SUBROUTINES. IT IS USED BY MAIN AND
 0702* SEGMENT PROGRAM TYPES. IT READS THE RELOCATABLE RECORDS FROM THE
 0703* SCRATCH PORTION OF THE DISC, AND WRITES THE ABSOLUTE CODE ON THE
 0704* USER FILE AREA OF THE DISC.
 0705*

0706	03005	000000	LOADS	NOP	
0707	03006	003400		CCA	
0708	03007	073420R		STA PLFLG	SET NO DBL RECORDS IN FLAG
0709	03010	002400	LOADN	CLA	INITIALLY SET FOR PASS 1
0710	03011	073417R	LD3	STA L001	PASS 2 WHEN LENGTH OF LOW LINK F
0711	03012	003400		CCA	SET FOR UPPER LINK
0712	03013	073412R		STA LWH1	AREA PRESENT
0713	03014	067417R		LDB L001	IF PASS 1,
0714	03015	006002		SZB	SAVE PROGRAM & BASE PAGE
0715	03016	027023R		JMP LD1	STARTING RELOCATION ADDRESSES
0716	03017	063603R		LDA TPREL	
0717	03020	073415R		STA LWH4	
0718	03021	063441R		LDA TBREL	
0719	03022	073414R		STA LWH3	
0720	03023	063177R	LD1	LDA RSS	RESTORE SKIP INSTRUCTION
0721	03024	073332R		STA BLIN3	
0722	03025	002400		CLA	SET FOR PASS 1 OR 2
0723	03026	073420R	LOADX	STA L01	PASS 3 WHEN LOW LINK OUT TO DISC
0724	03027	063417R		LDA L001	IF PASS 2 OR 3 (IE. SOME LOADING
0801	03030	002003		SZA,RSS	HAS BEEN PERFORMED), CLEAR
0802	03031	027035R		JMP LD2	BASE PAGE LINK ADDRESSES IN
0803	03032	063414R		LDA LWH3	LST ENTRIES
0804	03033	067441R		LDB TBREL	
0805	03034	016277R		JSB CLRLT	
0806	03035	063415R	LD2	LDA LWH4	RESTORE ORIGINAL PROG RELOCATION
0807	03036	067420R		LDB L01	BASE, UNLESS PASS 3. IF PASS 3
0808	03037	006003		SZB,RSS	RELOCATION BASE HAS BEEN AD-
0809	03040	073603R		STA TPREL	JUSTED PAST LOW LINK AREA.
0810	03041	063414R		LDA LWH3	RESTORE ORIGINAL BASE PAGE
0811	03042	073441R		STA TBREL	RELOCATION BASE.
0812	03043	016437R		JSB CLIST	BLANK MEMORY MAP BUFFER
0813	03044	002400		CLA	
0814	03045	167317R		LDB ID7,I	
0815	03046	006020		SSB	
0816	03047	002004		INA	
0817	03050	073443R		STA TDIR	
0818	03051	063560R		LDA AMLST	ADDR OF MEMORY MAP BUFFER

```

0019 03052 073321R STA AMAD
0020 03053 063374R LDA HDFLG CHECK IF MAIN OR SEGMENT LOADING
0021 03054 073411R STA TEMP2
0022 03055 002021 SSA,RSS
0023 03056 037321R ISZ AMAD
0024 03057 163311R LDA ID1,I STORE NAME INTO MEMORY MAP
0025 03060 173321R STA AMAD,I
0026 03061 037321R ISZ AMAD
0027 03062 073464R STA ZBUF
0028 03063 163312R LDA ID2,I
0029 03064 173321R STA AMAD,I
0030 03065 037321R ISZ AMAD
0031 03066 073465R STA ZBUF+1
0032 03067 163313R LDA ID3,I
0033 03070 010075 AND M1774
0034 03071 073466R STA ZBUF+2
0035 03072 033453R IOR BLANK
0036 03073 173321R STA AMAD,I
0037*
0038* DISSET INTERROGATES BIT 15 OF WD 4 OF THE
0039* IDENT TABLE,
0040* BIT15 = 1; FLAG DISCX IS SET(USER AREA)
0041* BIT15 = 0; FLAG DISCX IS CLEARED(SYSTEM AREA)
0042*
0043 03074 064056 LDB .3
0044 03075 163314R LDA ID4,I
0045 03076 002021 SSA,RSS
0046 03077 064055 LDB .2
0047 03100 077352R STB DISCX
0048*
0049 03101 063420R LDA L01 PASS 3?
0050 03102 002002 SZA
0051 03103 027107R JMP LD4 YES - TEST REAL FLAG
0052 03104 037411R ISZ TEMP2 NO - TEST TEMPORARY HDFLG
0053 03105 027153R JMP SUBHD
0054 03106 027111R JMP LD5
0055 03107 037374R LD4 ISZ HDFLG TEST REAL HDFLG
0056 03110 027153R JMP SUBHD
0057*
0058 03111 063354R LD5 LDA DSKAD GET CURRENT DISC ADDRESS
0059 03112 067420R LDB L01 IF NOT PASS 3,
0060 03113 006003 SZB,RSS
0061 03114 073360R STA DSKMN SAVE
0062 03115 063425R LDA PTYPE CHECK PROGRAM TYPE
0063 03116 050060 CPA .5
0064 03117 027124R JMP BGSPL SEGMENT SO CLEAR PLIST
0065 03120 063547R LDA N128
0066 03121 073417R STA PLCNT SET PLIST WORD COUNT
0067 03122 067564R LDB PLAD
0068 03123 027133R JMP CPLST CLEAR PROGRAM LIST
0069 03124 063564R BGSPL LDA PLAD
0070 03125 003004 CMA,INA
0071 03126 043422R ADA PLMAN
0072 03127 043547R ADA N128
0073 03130 073417R STA PLCNT
0074 03131 067422R LDB PLMAN

```

0075	03132	027141R		JMP CPXXX	
0076	03133	063311R	CPLST	LDA ID1	GET IDENT ADDR OF MAIN AND SET
0077	03134	170001		STA B,I	INTO PLIST
0078	03135	006004		INB	INCR CURRENT PLIST ADDRESS
0079	03136	037417R		ISZ PLCNT	SKIP=PLIST FULL
0080	03137	002001		RSS	
0081	03140	027146R		JMP SETMP	NO PLIST CLEARING REQUIRED
0082	03141	002400	CPXXX	CLA	
0083	03142	170001		STA B,I	CLEAR PLIST WORD
0084	03143	006004		INB	
0085	03144	037417R		ISZ PLCNT	
0086	03145	027142R		JMP *-3	
0087	03146	063415R	SETMP	LDA LWH4	
0088	03147	073427R		STA SV1	SAVE STARTING PROGRAM BOUND
0089	03150	063414R		LDA LWH3	
0090	03151	073430R		STA SV2	SAVE STARTING BASE PAGE BOUND
0091	03152	027161R		JMP OUTLN	OUTPUT NAME ONLY IF MAIN/SEGMENT
0092	03153	063415R	SUBHD	LDA LWH4	
0093	03154	067554R		LDB AMEM5	
0094	03155	016371R		JSB CONVD	CONVERT LOAD ADDRESS TO ASCII
0095	03156	063414R		LDA LWH3	
0096	03157	067556R		LDB AME12	
0097	03160	016371R		JSB CONVD	CONVERT SUBR BP STARTING ADDRESS
0098					
0099	03161	063420R	OUTLN	LDA L01	PASS 3?
0100	03162	002003		SZA,RSS	
0101	03163	027167R		JMP LD6	NO= SKIP MEMORY MAP
0102	03164	063437R		LDA ,36	
0103	03165	067560R		LDB AMLST	
0104	03166	016364R		JSB DRKEY	OUTPUT LINE TO LIST UNIT
0105	03167	163315R	LD6	LDA ID5,I	
0106	03170	016727R		JSB INNAM	INPUT NAM RECORD
0107	03171	063340R		LDA CURAL	GET RECORD TYPE
0108	03172	002004		INA	
0109	03173	160000		LDA A,I	
0110	03174	001723		ALF,RAR	
0111	03175	010002		AND ,7	
0112	03176	050054		CPA ,1	NAME RECORD?
0113	03177	002001	RSS	RSS	
0114	03200	026641R		JMP NMERR	
0115	03201	063340R		LDA CURAL	
0116	03202	040061		ADA ,6	
0117	03203	160000		LDA A,I	
0118	03204	073421R		STA PLGTH	SAVE PROGRAM LENGTH
0119	03205	017165R		JSB ALINK	SET UP LOW LINK AREA FOR CP LINK
0120	03206	067603R		LDB TPREL	
0121	03207	077427R		STB RELAD	
0122	03210	063603R		LDA LIBRY	SET CURRENT RELOCATION ADDRESS
0123	03211	002002		SZA	
0124	03212	027226R		JMP BPN	
0125	03213	063340R		LDA CURAL	
0126	03214	040064		ADA ,9	
0127	03215	160000		LDA A,I	
0128	03216	010073		AND M177	
0129	03217	050061		CPA ,6	
0130	03220	027224R		JMP SUBH5	

0131	03221	050062	CPA .7	
0132	03222	002001	RSS	
0133	03223	027226R	JMP BPN	
0134	03224	063603R	SUBHS LDA TPREL	
0135	03225	073603R	STA LIBRY	
0136	03226	163340R	BPN LDA CURAL,I	UPDATE TO END OF NAM RECORED
0137	03227	001727	ALF,ALF	
0138	03230	010074	AND M377	
0139	03231	070001	STA B	
0140	03232	063340R	LDA CURAL	
0141	03233	040001	ADA B	
0142	03234	073340R	STA CURAL	SET TO END OF NAM RECORD
0143	03235	063403R	LDA LCNT	
0144	03236	040001	ADA B	
0145	03237	073403R	STA LCNT	SET NEW CURRENT COUNT
0146	03240	016721R	CLSRC JSB DBSET	
0147	03241	163340R	LDA CURAL,I	
0148	03242	064000	LDB A	
0149	03243	001723	ALF,RAR	
0150	03244	010062	AND .7	
0151	03245	050055	CPA .2	
0152	03246	027513R	JMP DENTR	GO TO ENT RECORD PROCESSOR
0153	03247	050056	CPA .3	
0154	03250	027713R	JMP DBLR	GO TO DBL RECORD PROCESSOR
0155	03251	050057	CPA .4	
0156	03252	027514R	JMP DEXTR	GO TO EXT RECORD PROCESSOR
0157	03253	050060	CPA .5	
0158	03254	027256R	JMP DENDR	GO TO END RECORD PROCESSOR
0159	03255	026641R	JMP NMERR	
0160				
0161		END RECORD PROCESSOR		
0162				
0163	03256	063420R	DENDR LDA L01	IF END DETECTED DURING PASS 2,
0164	03257	002002	SZA	SET LWH1=0 TO INDICATE NO
0165	03260	027263R	JMP LD7	UPPER LINK AREA, THEN OUTPUT
0166	03261	073412R	STA LWH1	LOW LINK AREA TO
0167	03262	027317R	JMP CROSS	DISC
0168	03263	163340R	LD7 LDA CURAL,I	
0169	03264	010062	AND .7	SAVE TRANSFER & RELOCATION INDIC
0170	03265	073310R	STA PENTZ	
0171	03266	016721R	JSB DBSET	
0172	03267	016721R	JSB DBSET	
0173	03270	037427R	ISZ RELAD	
0174	03271	067374R	LDB HDFLG	ONLY WORRY ABOUT TRANSFER ADDR
0175	03272	006002	SZB	IF PROGRAM IS A MAIN OR SEGMENT
0176	03273	027317R	JMP DEN20	
0177	03274	067310R	LDB PENTZ	TRANSFER ADDRESS PRESENT?
0178	03275	004010	SLB	
0179	03276	027311R	JMP DEN30	YES- COMPUTE RELOC. TRANS ADDR
0180	03277	063421R	LDA PLGTH	NO- CHECK PROGRAM LENGTH
0181	03300	002003	SZA,RSS	ZERO LENGTH (LIBRARY)?
0182	03301	027316R	JMP DEN10	YES- IGNORE L08 ERROR
0183	03302	063460R	LDA UMFWA	SET PRIMARY ENTRY FOR DRIVER
0184	03303	067462R	LDB DVR	IS THIS TYPE 4
0185	03304	054057	CPB .4	DEVICE DRIVER?
0186	03305	027316R	JMP DEN10	YES - IGNORE L08 ERROR

0187	03306	063476R		LDA ERR08	NO SO OUTPUT ERROR MESSAGE
0188	03307	026631R		JMP ABRTX	ABORT LOADR
0189	03310	000000	PENTZ	NOP	PRIMARY ENTRY PRESENT FLAG
0190					
0191	03311	005100	DEN30	BRS	REMOVE TRANSFER ADDR. INDICATOR
0192	03312	047601R		ADB RBTAD	
0193	03313	006004		INB	
0194	03314	160001		LDA B,I	GET PROPER RELOCATION BASE
0195	03315	143340R		ADA CURAL,I	ADD RELOC VALUE FROM END RECORD
0196	03316	073424R	DEN10	STA PRENT	SAVE PRIMARY ENTRY POINT
0197	03317	016151R	DEN20	JSB DEBUG	
0198	03320	016230R		JSB INLST	
0199	03321	016234R	CLST	JSB LSTX	SET LST ADDRESS
0200	03322	027327R		JMP LSTCR	END OF LST
0201	03323	163302R		LDA LST3,I	
0202	03324	010076		AND M1774	CLEAR LST ORDINAL
0203	03325	173302R		STA LST3,I	
0204	03326	027321R		JMP CLST	CONTINUE CLEARING ORDINAL
0205	03327	063421R	LSTCR	LDA PLGTH	GET PROGRAM LENGTH
0206	03330	001265		RAL,CLE,ERA	SET E=SIGN
0207	03331	043603R		ADA TPREL	ADD PROGRAM RELOCATION BASE
0208	03332	002040		SEZ	SKIP IF ASSEMBLER PRODUCED
0209	03333	063427R		LDA RELAD	GET CURRENT RELOCATION ADDRESS
0210	03334	073603R		STA TPREL	SET NEW RELOCATION BASE
0211	03335	070001		STA B	
0212	03336	063427R		LDA RELAD	GET LAST WORD ADDRESS
0213	03337	003004		CMA,INA	
0214	03340	044000		ADB A	ADD RELOCATION BASE
0215	03341	006003		SZ0,RSS	
0216	03342	027346R		JMP **4	
0217	03343	002400		CLA	
0218	03344	067603R		LDB TPREL	
0219	03345	017706R		JSB LABDO	
0220	03346	063420R		LDA L01	IF PASS 3,
0221	03347	002021		SSA,RSS	OUTPUT UPPER
0222	03350	017356R		JSB BLINK	LINK AREA
0223	03351	063547R		LDA N128	
0224	03352	073417R		STA PLCNT	SET PROG LIST COUNTER
0225	03353	067564R		LDB PLAD	GET ADDR OF PROGRAM LIST
0226	03354	160001	PLSCN	LDA B,I	GET PROGRAM LIST ENTRY
0227	03355	002003		SZA,RSS	SKIP-ENTRY IN PLIST
0228	03356	027436R		JMP DEIDG	
0229	03357	002020		SSA	SKIP IF PROG LOADED
0230	03360	027475R		JMP ENTID	GET NEXT PROG IDENT ADDRESS
0231	03361	006004	EPLST	INB	INCR PROGRAM LIST ADDRESS
0232	03362	037417R		ISZ PLCNT	SKIP IF END OF PROGRAM LIST
0233	03363	027354R		JMP PLSCN	NO-CONTINUE PROG LIST SEARCH
0234	03364	016355R	CLFLG	JSB SPACE	SKIP A LINE
0235	03365	016437R		JSB CLIST	CLEAR OUTPUT BUFFER
0236	03366	067431R		LDB BNDHD	PUT "(BOUNDS)" IN OUTPUT BUFFER
0237	03367	160001		LDA B,I	
0238	03370	073253R		STA MLIST	
0239	03371	006004		INB	
0240	03372	160001		LDA B,I	
0241	03373	073254R		STA MLIST+1	
0242	03374	006004		INB	

0243	03375	160001	LDA B,I	
0244	03376	073255R	STA MLIST+2	
0245	03377	006004	INB	
0246	03400	160001	LDA B,I	
0247	03401	073256R	STA MLIST+3	
0248	03402	063427R	LDA SV1	
0249	03403	067554R	LDB AMEM5	
0250	03404	016371R	JSB CONVD	CONVERT STARTING PROGRAM BOUND
0251	03405	063430R	LDA SV2	
0252	03406	067556R	LDB AME12	
0253	03407	016371R	JSB CONVD	CONVERT STARTING BASE PG BOUND
0254	03410	063603R	LDA TPREL	
0255	03411	067555R	LDB AMEM8	
0256	03412	016371R	JSB CONVD	CONVERT ENDING PROGRAM BOUND
0257	03413	063441R	LDA TBREL	
0258	03414	067557R	LDB AME15	
0259	03415	016371R	JSB CONVD	CONVERT ENDING BASE PG BOUND
0260	03416	063437R	LDA ,36	
0261	03417	067560R	LDB AMLST	
0262	03420	016364R	JSB DRKEY	OUTPUT BOUNDS TO LIST DEVICE
0263	03421	016355R	JSB SPACE	
0264*				
0265	03422	063425R	LDA PTYPE	GET PROGRAM TYPE
0266	03423	050056	CPA ,3	USER MAIN
0267	03424	127005R	JMP LOADS,I	RETURN
0268	03425	016527R	JSB CLID3	CLEAR PROG LOADED FLAGS
0269	03426	127005R	JMP LOADS,I	RETURN
0270	03427	000000	SV1 NOP	
0271	03430	000000	SV2 NOP	
0272	03431	003432R	BNDHD DEF ++1	
0273	03432	024102	ASC 4,(BOUNDS)	
0274	03436	063344R	DEIDG LDA DBFLG	CHECK IF DEBUG TO BE LOADED
0275	03437	002003	SZA,RSS	
0276	03440	027364R	JMP CLFLG	NO
0277	03441	063333R	LDA CLXXX	CHECK IF ALREADY LOADED
0278	03442	002002	SZA	
0279	03443	027364R	JMP CLFLG	YES
0280	03444	037333R	ISZ CLXXX	INCR DEBUG LOADED FLAG
0281	03445	077444R	STB TEMP	
0282	03446	016230R	JSB INLST	SEARCH FOR DEBUG IN LST
0283	03447	016234R	DEI10 JSB LSTX	
0284	03450	026637R	JMP RCERR	
0285	03451	163300R	LDA LST1,I	
0286	03452	053462R	CPA CHRDE	
0287	03453	002001	RSS	
0288	03454	027447R	JMP DEI10	TRY NEXT LST
0289	03455	163301R	LDA LST2,I	
0290	03456	053463R	CPA CHRDE+1	
0291	03457	002001	RSS	
0292	03460	027447R	JMP DEI10	TRY NEXT LST
0293	03461	163302R	LDA LST3,I	
0294	03462	010075	AND M1774	
0295	03463	070001	STA B	
0296	03464	063464R	LDA CHRDE+2	
0297	03465	010075	AND M1774	
0298	03466	050001	CPA B	

0299	03467	002001		RSS	DEBUG FOUND
0300	03470	027447R		JMP DEI10	TRY NEXT ENTRY
0301	03471	067444R		LDB TEMP	
0302	03472	163303R		LDA LST4,I	
0303	03473	033450R		IOR MSIGN	
0304	03474	170001		STA B,I	SET IDENT ADDRESS IN PLIST TABLE
0305	03475	001265	ENTID	RAL,CLE,ERA	CLEAR SIGN(PROG TO BE LOADED)
0306	03476	073306R		STA TIDNT	SET IDENT ADDR FOR IDX
0307	03477	072210R		STA TBUF	SAVE CURRENT IDENT ADDRESS
0308	03500	077341R		STB CURAP	SAVE CURRENT PLIST ADDRESS
0309	03501	016265R		JSB IDX	SET IDENT ADDRESSES
0310	03502	026637R		JMP RCERR	
0311	03503	067341R		LDB CURAP	GET CURRENT PLIST ADDRESS
0312	03504	163313R		LDA ID3,I	GET PROGRAM USAGE FLAG
0313	03505	002014		SLA,INA	SKIP-PROG NOT YET LOADED
0314	03506	027361R		JMP EPLST	TEST FOR END OF PLIST
0315	03507	173313R		STA ID3,I	SET PROG USAGE FLAG=LOADED
0316	03510	062210R		LDA TBUF	GET CURRENT IDENT ADDRESS
0317	03511	170001		STA B,I	SET ENTRY IN PROG LIST
0318	03512	027010R		JMP LOADN	LOAD NEXT PROGRAM
0319*					
0320*	ENT/EXT RECORD PROCESSOR				
0321*					
0322	03513	003401	DENTR	CCA,RSS	SET ENT FLAG
0323	03514	002400	DEXTR	CLA	SET EXT FLAG
0324	03515	073410R		STA NXFLG	SAVE ENT/EXT FLAG
0325	03516	060001		LDA B	
0326	03517	010071		AND M37	
0327	03520	003004		CMA,INA	
0328	03521	073371R		STA EXCNT	SET SYMBOL COUNTER
0329	03522	016721R		JSB DBSET	GET ADDR OF NEXT WORD IN LBUF
0330	03523	016721R		JSB DBSET	GET ADDR OF NEXT WORD IN LBUF
0331	03524	163340R	NXSYM	LDA CURAL,I	
0332	03525	072210R		STA TBUF	SAVE NAME 1,2
0333	03526	016721R		JSB DBSET	
0334	03527	163340R		LDA CURAL,I	
0335	03530	072211R		STA TBUF+1	SAVE NAME 3,4
0336	03531	016721R		JSB DBSET	
0337	03532	163340R		LDA CURAL,I	
0338	03533	072212R		STA TBUF+2	SAVE NAME 5
0339	03534	016721R		JSB DBSET	
0340	03535	016230R		JSB INLST	INITIALIZE LSTX
0341	03536	016234R	ENEX1	JSB LSTX	SET LST ADDRESSES
0342	03537	027560R		JMP XXXX1	
0343	03540	062210R		LDA TBUF	
0344	03541	153300R		CPA LST1,I	NAME 1,2 EQUAL?
0345	03542	002001		RSS	YES
0346	03543	027536R		JMP ENEX1	NO-TRY NEXT LST ENTRY
0347	03544	062211R		LDA TBUF+1	
0348	03545	153301R		CPA LST2,I	NAME 3,4 EQUAL?
0349	03546	002001		RSS	YES
0350	03547	027536R		JMP ENEX1	NO-TRY NEXT LST ENTRY
0351	03550	062212R		LDA TBUF+2	
0352	03551	010075		AND M1774	
0353	03552	070001		STA B	
0354	03553	163302R		LDA LST3,I	UPPER CHAR OF

0355	03554	010075	AND M1774	NAME 5 EQUAL?
0356	03555	050001	CPA B	
0357	03556	002001	RSS	YES
0358	03557	027536R	JMP ENEX1	NO=TRY NEXT LST ENTRY
0359	03560	063410R	XXXXX1 LDA NXFLG	
0360	03561	002003	SZA, RSS	SKIP IF ENT
0361	03562	027625R	JMP EXT2	GO TO PROCESS EXT
0362	03563	062212R	LDA TBUF+2	
0363	03564	010056	AND .3	
0364	03565	043601R	ADA RBTAD	
0365	03566	002004	INA	
0366	03567	164000	LDB A, I	GET CURRENT RELOCATION BASE
0367	03570	147340R	ADB CURAL, I	ADD CURRENT RELOC VALUE
0368	03571	077411R	STB OPRND	SAVE ABS ENT PT ADDRESS
0369	03572	163304R	LDA LST5, I	GET BP LINK ADDRESS
0370	03573	002003	SZA, RSS	SKIP=BP LINK ADDR SET
0371	03574	016601R	JSB BPSCN	SET BP LINK ADDR, BP LINK
0372	03575	173304R	STA LST5, I	SET BP LINK ADDR IN LST
0373	03576	016671R	JSB SETOP	SET OPERAND IN BASE PAGE
0374	03577	063420R	LDA L01	UNLESS IT'S
0375	03600	002003	SZA, RSS	PASS 3, SKIP
0376	03601	027623R	JMP NLENT	ENT LIST
0377	03602	016437R	JSB CLIST	CLEAR MEMORY MAP BUFFER
0378	03603	063451R	LDA BLAST	
0379	03604	073254R	STA MLIST+1	
0380	03605	163300R	LDA LST1, I	
0381	03606	073255R	STA MLIST+2	SET NAME 1,2 IN MEMORY MAP
0382	03607	163301R	LDA LST2, I	
0383	03610	073256R	STA MLIST+3	SET NAME 3,4 IN MEMORY MAP
0384	03611	163302R	LDA LST3, I	
0385	03612	010075	AND M1774	
0386	03613	033453R	IOR BLANK	
0387	03614	073257R	STA MLIST+4	SET NAME 5 IN MEMORY MAP
0388	03615	063411R	LDA OPRND	GET ABSOLUTE ENTRY PT ADDRESS
0389	03616	067554R	LDB AMEM5	
0390	03617	016371R	JSB CONV D	CONVERT TO ASCII
0391	03620	063435R	LDA .16	
0392	03621	067560R	LDB AMLST	
0393	03622	016364R	JSB DRKEY	PRINT ENTRY POINT
0394	03623	016721R	NLENT JSB DBSET	GET ADDR OF NEXT WORD IN LBUF
0395	03624	027710R	JMP EXEND	GO PROCESS NEXT SYMBOL
0396	03625	062212R	EXT2 LDA TBUF+2	GET ORDINAL
0397	03626	173302R	STA LST3, I	SET ORDINAL IN LST
0398	03627	163304R	LDA LST5, I	GET BP LINK ADDRESS
0399	03630	002003	SZA, RSS	SKIP=BP LINK ADDRESS SET
0400	03631	016702R	JSB ALLOC	ALLOCATE NEW BP LOCATION
0401	03632	173304R	STA LST5, I	SET NEW BP LINK ADDRESS
0402	03633	163303R	LDA LST4, I	GET IDENT ADDRESS
0403	03634	002002	SZA	
0404	03635	002020	SSA	
0405	03636	027710R	JMP EXEND	IGNORE UNDEFINED PROGRAM CALL
0406	03637	073306R	STA TIDNT	SET ID ADDR FOR IDX
0407	03640	063311R	LDA ID1	GET CURRENT IDENT ADDR
0408	03641	072210R	STA TBUF	SAVE CURRENT IDENT ADDRESS
0409	03642	016265R	JSB IDX	
0410	03643	027710R	JMP EXEND	

0411	03644	163316R	LDA ID6,I	GET M/S, TYPE
0412	03645	072211R	STA TBUF+1	
0413	03646	163313R	LDA ID3,I	GET PROGRAM USEAGE FLAG
0414	03647	072212R	STA TBUF+2	
0415	03650	062210R	LDA TBUF	GET CURRENT IDENT ADDRESS
0416	03651	073306R	STA TIDNT	
0417	03652	016265R	JSB IDX	
0418	03653	027710R	JMP EXEND	
0419	03654	062211R	LDA TBUF+1	GET M/S, TYPE FOR EXT
0420	03655	001265	RAL,CLE,ERA	SET E=M/S
0421	03656	010073	AND M177	ISOLATE TYPE
0422	03657	050061	CPA ,6	
0423	03660	027665R	JMP LIBUT	LIBRARY-TEST FOR LOADING
0424	03661	050062	CPA ,7	
0425	03662	027665R	JMP LIBUT	
0426	03663	002040	SEZ	SKIP-NOT MAIN PROGRAM
0427	03664	027710R	JMP EXEND	IGNORE PROGRAM CALL
0428	03665	063547R	LIBUT LDA N128	
0429	03666	073417R	STA PLCNT	SET PROGRAM LIST COUNTER
0430	03667	067564R	LDB PLAD	GET PROGRAM LIST ADDRESS
0431	03670	160001	PLEXT LDA B,I	GET PROG LIST ENTRY
0432	03671	002003	SZA,RSS	SKIP IF IDENT ADDRESS
0433	03672	027702R	JMP SEXT	MAKE ENTRY IN PROG LIST
0434	03673	001265	RAL,CLE,ERA	CLEAR SIGN
0435	03674	153303R	CPA LST4,I	PROG PREVIOUSLY CALLED?
0436	03675	027710R	JMP EXEND	YES=IGNORE EXT SYMBOL
0437	03676	006004	INB	INCR PROG LIST ADDR
0438	03677	037417R	ISZ PLCNT	END OF PROG LIST?
0439	03700	027670R	JMP PLEXT	NO-GO GET NEXT PROGRAM
0440	03701	027710R	JMP EXEND	TEST FOR ALL EXT PROCESSED
0441	03702	062212R	SEXT LDA TBUF+2	GET PROGRAM USEAGE FLAG
0442	03703	000010	SLA	SKIP-PROGRAM NOT LOADED
0443	03704	027710R	JMP EXEND	OMIT PROGRAM LIST ENTRY
0444	03705	163303R	LDA LST4,I	GET CALLED PROG IDENT ADDR
0445	03706	033450R	IQR MSIGN	SET SIGN TO SHOW NOT LOADED
0446	03707	170001	STA B,I	SET IN PROGRAM LIST
0447	03710	037371R	EXEND ISZ EXCNT	SKIP-ALL SYMBOLS PROCESSED
0448	03711	027524R	JMP NXSYM	NO-PROCESS NEXT SYMBOL
0449	03712	027240R	JMP CLSRC	NO-CLASSIFY NEXT RECORD
0450*				
0451*	DBL RECORD PROCESSOR			
0452*				
0453	03713	060001	DDBLR LDA B	
0454	03714	010072	AND M77	
0455	03715	003004	CMA,INA	
0456	03716	073371R	STA EXCNT	SET INSTRUCTION COUNT
0457	03717	016721R	JSB DBSET	
0458	03720	016721R	JSB DBSET	
0459	03721	167340R	LDB CURAL,I	GET RELOCATION ADDRESS
0460	03722	077346R	STB DBLAD	SAVE RELOCATION ADDRESS
0461	03723	037420R	ISZ PLFLG	SKIP-FIRST DBL RECORD
0462	03724	027737R	JMP DBL0	IGNORE SUBSEQUENT RECORDS
0463	03725	063420R	LDA L01	
0464	03726	002002	SZA	
0465	03727	163316R	LDA ID6,I	GET TYPE
0466	03730	010073	AND M177	

0467	03731	050060	CPA .5	
0468	03732	002001	RSS	SET NEW MEMORY BOUND FOR SEGMENT
0469	03733	006400	CLB	SET PGMAD=0
0470	03734	077326R	HABSS STB BSSDP	SAVE INITIAL PROG DISPLACEMENT
0471	03735	047415R	ADB LWH4	
0472	03736	077332R	STB CBASE	SAVE CURRENT BASE PROGRAM ADDRESS
0473	03737	016721R	DBL0 JSB DBSET	
0474	03740	167340R	DBL1 LDB CURAL,I	GET RELOCATION BYTES
0475	03741	077430R	STB REKEY	SAVE FOR RELOCATION TYPE
0476	03742	060046	LDA N5	
0477	03743	073401R	STA INSCN	SET RELOCATION BYTE COUNT
0478	03744	016721R	JSB DBSET	
0479	03745	063430R	DBL2 LDA REKEY	GET RELOCATION BYTES
0480	03746	001723	ALF,RAR	
0481	03747	073430R	STA REKEY	SAVE FOR NEXT INST WORD
0482	03750	010062	AND .7	ISOLATE CURRENT BYTE
0483	03751	050057	CPA .4	EXTERNAL REFERENCE?
0484	03752	026002R	JMP DBL4	YES-GET LINK ADDRESS
0485	03753	050060	CPA .5	MEMORY REFERENCE?
0486	03754	026023R	JMP DBL6	YES-CHECK FOR INDIRECT LINK
0487	03755	050061	CPA .6	SPECIAL REFERENCE?
0488	03756	026127R	JMP BYTE	YES- CHECK FOR BYTE ADDRESS
0489	03757	043601R	ADA RBTAD	
0490	03760	160000	LDA A,I	GET RELOCATION BASE
0491	03761	143340R	ADA CURAL,I	ADD CURRENT INSTRUCTION WORD
0492	03762	067346R	DBL3 LDB DBLAD	CURRENT RELOCATION BASE
0493	03763	047603R	ADB TPREL	ADD CURRENT PROG RELOCATION BASE
0494	03764	073411R	STA TEMP2	SAVE A-REGISTER
0495	03765	063420R	LDA L01	UNLESS IT'S
0496	03766	002003	SZA,RSS	PASS 3, DON'T
0497	03767	027772R	JMP LD8	OUTPUT TO DISC
0498	03770	063411R	LDA TEMP2	RESTORE A-REGISTER
0499	03771	017706R	JSB LABD0	OUTPUT ABSOLUTE WORD
0500	03772	016721R	LD8 JSB DBSET	
0501	03773	037371R	ISZ EXCNT	SKIP-LAST INSTRUCTION OUT
0502	03774	002001	RSS	NO-CONTINUE
0503	03775	027240R	JMP CLSRC	YES-CLASSIFY NEXT RECORD
0504	03776	037346R	ISZ DBLAD	INCR DBL RELOCATION ADDRESS
0505	03777	037401R	ISZ INSCN	SKIP IF NEW RELOCATION BYTE
0506	04000	027745R	JMP DBL2	PROCESS NEXT INSTRUCTION
0507	04001	027740R	JMP DBL1	GET NEXT RELOCATION BYTE
0508	04002	163340R	DBL4 LDA CURAL,I	GET CURRENT DBL WORD
0509	04003	010074	AND M377	
0510	04004	073372R	STA EXORD	ISOLATE AND SAVE ORDINAL
0511	04005	016230R	JSB INLST	INITIALIZE LSTX
0512	04006	016234R	DBL5 JSB LSTX	SET LST ADDRESSES
0513	04007	027762R	JMP DBL3	ORDINAL NOT FOUND
0514	04010	163302R	LDA LST3,I	GET WORD3 OF LST ENTRY
0515	04011	010074	AND M377	
0516	04012	053372R	CPA EXORD	SAME ORDINAL?
0517	04013	002001	RSS	YES
0518	04014	026006R	JMP DBL5	NO-CONTINUE SEARCH
0519	04015	167304R	LDB LST5,I	GET BP LINK ADDRESS
0520	04016	163340R	LDA CURAL,I	GET CURRENT DBL WORD
0521	04017	013447R	AND M1740	ISOLATE INSTRUCTION CODE
0522	04020	033450R	IOR MSIGN	SET INDIRECT

0523	04021	030001	IOR B	SET ADDRESS=BP LINK ADDRESS
0524	04022	027762R	JMP DBL3	OUTPUT INSTRUCTION
0525	04023	163340R DBL6	LDA CURAL,I	GET CURRENT DBL WORD
0526	04024	073402R	STA INSTR	SAVE INSTRUCTION CODE
0527	04025	016721R	JSB DBSET	GET ADDR OF NEXT WORD IN LBUF
0528	04026	063402R	LDA INSTR	GET INSTRUCTION CODE
0529	04027	010056	AND ,3	MASK RELOCATION INDICATOR
0530	04030	043601R	ADA RBTAD	
0531	04031	002004	INA	
0532	04032	164000	LDB A,I	GET CURRENT RELOCATION BASE
0533	04033	147340R	ADB CURAL,I	ADD CURRENT INSTR ADDRESS
0534	04034	076126R	STB TMPAD	AND SAVE TEMPORARILY
0535	04035	063402R	LDA INSTR	
0536	04036	001323	RAR,RAR	
0537	04037	010074	AND M377	
0538	04040	002003	SZA,RSS	EXT ORDINAL IN BITS 2-9 ?
0539	04041	026064R	JMP TSTIN	NO- CONTINUE
0540	04042	073372R	STA EXORD	YES- SAVE FOR COMPARE
0541	04043	016230R	JSB INLST	SEARCH FOR EXT
0542	04044	016234R SREXT	JSB LSTX	
0543	04045	026124R	JMP UND	(SHOULD FIND IT)
0544	04046	163302R	LDA LST3,I	
0545	04047	010074	AND M377	
0546	04050	053372R	CPA EXORD	MATCH?
0547	04051	002001	RSS	YES-SKIP
0548	04052	026044R	JMP SREXT	NO- TRY AGAIN
0549	04053	163303R	LDA LST4,I	IS ENTRY DEFINED?
0550	04054	002002	SZA	
0551	04055	002020	SSA	
0552	04056	026124R	JMP UND	NO - TREAT AS NO EXT
0553	04057	163304R	LDA LST5,I	YES- FIND ABSOLUTE ENT VALUE
0554	04060	043331R	ADA BPOIT	
0555	04061	043511R	ADA BPBUF	
0556	04062	164000	LDB A,I	
0557	04063	046126R	ADB TMPAD	ADD SAVED WORD 2 VALUE
0558	04064	063402R TSTIN	LDA INSTR	GET ORIGINAL WORD 1
0559	04065	013447R	AND M1740	
0560	04066	002020	SSA	SKIP IF DIRECT REFERENCE
0561	04067	047450R	ADB MSIGN	SET INDIRECT BIT
0562	04070	002003	SZA,RSS	ADDRESSABLE INSTRUCTION?
0563	04071	026122R	JMP DBL9	NO- SKIP PAGE CHECKING
0564	04072	073402R	STA INSTR	SAVE INSTRUCTION CODE
0565	04073	063603R	LDA TPREL	
0566	04074	043346R	ADA DBLAD	
0567	04075	013446R	AND M0760	ISOLATE PAGE #
0568	04076	073412R	STA PAGNO	SAVE CURRENT PAGE #
0569	04077	060001	LDA B	GET INSTRUCTION ADDRESS
0570	04100	013446R	AND M0760	ISOLATE PAGE #
0571	04101	053412R	CPA PAGNO	SAME PAGE?
0572	04102	026115R	JMP DBL7	YES-NO INDIRECT ADDRESS NEEDED
0573	04103	077411R	STB OPRND	SAVE INSTRUCTION ADDRESS
0574	04104	016601R	JSB BPSCN	SCAN BASE PAGE FOR SAME OPERAND
0575	04105	016671R LINKC	JSB SETOP	SET OPERAND IN BP LOCATION
0576	04106	070001	STA B	SAVE LINK ADDRESS
0577	04107	013444R	AND 01777	GET PAGE OFFSET
0578	04110	050001	CPA B	CURRENT PAGE?

0579	04111	002001		RSS	NO- SKIP
0580	04112	033445R		IOR M2000	SET Z/C = 1 (CURRENT)
0581	04113	033450R		IOR MSIGN	SET INDIRECT BIT
0582	04114	026120R		JMP DBL8	GO ADD INSTRUCTION & STORE WORD
0583	04115	060001	DBL7	LDA B	GET OPERAND ADDRESS
0584	04116	013444R		AND 01777	
0585	04117	033445R		IOR M2000	SET Z/C=1 (CURRENT)
0586	04120	033402R	DBL8	IOR INSTR	ADD INSTRUCTION CODE
0587	04121	027762R		JMP DBL3	OUTPUT INSTRUCTION WORD
0588*					
0589	04122	060001	DBL9	LDA B	PUT CONFIGURED I/O INSTR IN "A"
0590	04123	027762R		JMP DBL3	OUTPUT INSTRUCTION
0591*					
0592	04124	066126R	UND	LDB TMPAD	
0593	04125	026064R		JMP TSTIN	
0594*					
0595	04126	000000	TMPAD	NOP	
0596*					
0597*	R=6			SPECIAL REFERENCE	
0598*					
0599	04127	163340R	BYTE	LDA CURAL,I	IS TYPE = 0? (IE BYTE ADDRESS)
0600	04130	001700		ALF	
0601	04131	010066		AND ,17	
0602	04132	002002		SZA	
0603	04133	026147R		JMP NBYT	NO- IGNORE INSTRUCTION
0604*					
0605	04134	163340R		LDA CURAL,I	GET RELOCATION INDICATOR
0606	04135	010056		AND ,3	
0607	04136	043601R		ADA RBTAD	
0608	04137	002004		INA	
0609	04140	160000		LDA A,I	GET RELOCATION BASE FROM TABLE
0610	04141	001000		ALS	OFFSET TO BYTE ADDRESS BASE
0611	04142	072126R		STA TMPAD	SAVE
0612	04143	016721R		JSB DBSET	POINT TO RELOC BYTE ADDRESS
0613	04144	163340R		LDA CURAL,I	COMPUTE TOTAL BYTE ADDRESS
0614	04145	042126R		ADA TMPAD	(BASE + OFFSET)
0615	04146	027762R		JMP DBL3	OUTPUT INSTRUCTION WORD
0616*					
0617	04147	016721R	NBYT	JSB DBSET	BUMP TO 2ND WORD
0618	04150	027772R		JMP LDB	IGNORE INSTRUCTION


```

0620*      DEBUG CHECKS IF CURRENT PROGRAM LOADED WAS DEBUG.
0621*      THE PRIMARY ENTRY POINT OF DEBUG IS SET INTO PRENT, AND THE
0622*      ADDRESS OF DEBUG IS SET TO BE INDIRECT.
0623*      THE PROGRAM PRIMARY ENTRY POINT IS SET INTO DEBUF
0624 04151 000000  DEBUG NOP
0625 04152 063464R      LDA ZBUF      CHECK IF
0626 04153 053462R      CPA CHRDE      -DEBUG- PROGRAM
0627 04154 002001      RSS
0628 04155 126151R      JMP DEBUG,I  NO-RETURN
0629 04156 063465R      LDA ZBUF+1
0630 04157 053463R      CPA CHRDE+1
0631 04160 002001      RSS
0632 04161 126151R      JMP DEBUG,I  NO-RETURN
0633 04162 063466R      LDA ZBUF+2
0634 04163 053464R      CPA CHRDE+2
0635 04164 002001      RSS
0636 04165 126151R      JMP DEBUG,I  NO-RETURN
0637 04166 016230R      JSB INLST
0638 04167 016234R DSCAN JSB LSTX      FIND -DEBUG- ENTRY IN LST FOR
0639 04170 026637R      JMP RCERR
0640 04171 163300R      LDA LST1,I
0641 04172 053462R      CPA CHRDE
0642 04173 002001      RSS
0643 04174 026167R      JMP DSCAN
0644 04175 163301R      LDA LST2,I
0645 04176 053463R      CPA CHRDE+1
0646 04177 002001      RSS
0647 04200 026167R      JMP DSCAN
0648 04201 163302R      LDA LST3,I
0649 04202 010075      AND M1774
0650 04203 053464R      CPA CHRDE+2
0651 04204 002001      RSS
0652 04205 026167R      JMP DSCAN
0653 04206 037347R      ISZ DBUGF
0654 04207 163303R      LDA LST4,I  GET ENT FOR DEBUG
0655 04210 002020      SSA
0656 04211 026637R      JMP RCERR
0657 04212 167304R      LDB LST5,I  GET BP LINK ADDR OF -DEBUG-
0658 04213 047331R      ADB BPOIT
0659 04214 047511R      ADB BPBUF
0660 04215 077444R      STB TEMP      SAVE BASE PAGE LINK ADDRESS
0661 04216 164001      LDB B,I
0662 04217 063424R      LDA PRENT
0663 04220 017706R      JSB LABDO
0664 04221 163444R      LDA TEMP,I
0665 04222 033450R      IOR MSIGN
0666 04223 173444R      STA TEMP,I
0667 04224 163340R      LDA CURAL,I
0668 04225 043603R      ADA TPREL
0669 04226 073424R      STA PRENT
0670 04227 126151R      JMP DEBUG,I

```

```

0672*      INLST SETS THE ADDRESS OF THE FIRST ENTRY IN LST IN TLST
0673*      CALLING SEQUENCE
0674*      JSB INLST
0675 04230 000000 INLST NOP
0676 04231 063275R LDA BLST
0677 04232 073276R STA TLST
0678 04233 126230R JMP INLST,I
0679*
0680*      LSTX SETS THE CURRENT LST ADDR INTO LST 1-5
0681*      CALLING SEQUENCE
0682*      JSB LSTX
0683*      RETURN
0684*      (P+1) IF AT END OF LST
0685*      (P+2) IF NOT AT END OF LST
0686*
0687 04234 000000 LSTX NOP
0688 04235 063276R LDA TLST      CHECK IF AT END OF LOADER SYMBOL
0689 04236 063277R CPA PLST      TABLE LIST
0690 04237 002001 RSS
0691 04240 036234R ISZ LSTX      YES, SO INCR TO DONE EXIT
0692 04241 073300R STA LST1      SET LOADER SYMBOL TABLE ADDRESS
0693 04242 002004 INA
0694 04243 073301R STA LST2
0695 04244 002004 INA
0696 04245 073302R STA LST3
0001 04246 002004 INA
0002 04247 073303R STA LST4
0003 04250 002004 INA
0004 04251 073304R STA LST5
0005 04252 002004 INA
0006 04253 073276R STA TLST
0007 04254 003004 CMA,INA      CHECK IF IDENT AND LST TABLES
0008 04255 043307R ADA PIDNT      OVERLAP
0009 04256 002021 SSA,RSS
0010 04257 126234R JMP LSTX,I      RETURN
0011 04260 026630R JMP LOVER      SYMBOL TABLE OVERFLOW
0012*
0013*      INIDX SETS THE ADDRESS OF THE FIRST ENTRY IN THE PROGRAM
0014*      IDENT TABLE AS THE CURRENT ADDRESS.
0015*      CALLING SEQUENCE
0016*      JSB INIDX
0017 04261 000000 INIDX NOP
0018 04262 063305R LDA BIDNT
0019 04263 073306R STA TIDNT
0020 04264 126261R JMP INIDX,I
0021*      IDX SETS THE ADDRESSES OF THE CURRENT 6 WORD ENTRY IN THE
0022*      IDENT TABLE FROM THE ADDRESS OF THE CURRENT ENTRY
0023*      CALLING SEQUENCE :
0024*      JSB IDX
0025*      RETURN:
0026*      (P+1) IF AT END OF IDENT TABLE
0027*      (P+2) IF NOT AT END OF IDENT TABLE
0028 04265 000000 IDX NOP
0029 04266 063306R LDA TIDNT
0030 04267 053307R CPA PIDNT
0031 04270 002001 RSS

```

0032	04271	036265R	ISZ	IDX	
0033	04272	073311R	STA	ID1	
0034	04273	002004	INA		
0035	04274	073312R	STA	ID2	
0036	04275	002004	INA		
0037	04276	073313R	STA	ID3	
0038	04277	002004	INA		
0039	04300	073314R	STA	ID4	
0040	04301	002004	INA		
0041	04302	073315R	STA	ID5	
0042	04303	002004	INA		
0043	04304	073316R	STA	ID6	
0044	04305	002004	INA		
0045	04306	073317R	STA	ID7	
0046	04307	063306R	LDA	TIDNT	
0047	04310	067306R	LDB	TIDNT	
0048	04311	003004	CMA,	INA	
0049	04312	043277R	ADA	PLST	
0050	04313	002021	SSA,	RS9	
0051	04314	026320R	JMP	++4	
0052	04315	044044	ADB	N7	
0053	04316	077306R	STB	TIDNT	
0054	04317	126265R	JMP	IDX,I	
0055	04320	063404R	LDA	LDING	CHECK=PROGRAM LOADING FLAG
0056	04321	002002	SZA		
0057	04322	026326R	JMP	++4	
0058	04323	063501R	LDA	ERR11	
0059	04324	016344R	JSB	ERROR	
0060	04325	026632R	JMP	ABORT	
0061	04326	003400	CCA		LOADING SO RETURN TO P+1
0062	04327	042265R	ADA	IDX	FOR PROGRAM PROCESSING
0063	04330	124000	JMP	A,I	

```

0065*      SYOUT PRINTS MESSAGE ON SYSTEM TELETYPE, THESE INCLUDE ALL
0066*      ERROR DIAGNOSTICS AND OPERATOR ACTION MESSAGES,
0067*      CALL SEQUENCE:
0068*      LDA <# CHARACTERS IN MESSAGE (+)>
0069*      LDB <MESSAGE ADDRESS>
0070*      JSB SYOUT
0071 04331 000000 SYOUT NOP
0072 04332 076341R STB SYM
0073 04333 003004 CMA,INA
0074 04334 073447R STA TTYNO
0075 04335 016001X JSB EXEC
0076 04336 004343R DEF **5
0077 04337 000055 DEF .2
0078 04340 000054 DEF .1
0079 04341 004341R SYM DEF *
0080 04342 011447R DEF TTYNO
0081 04343 126331R JMP SYOUT,I RETURN
0082*      ERROR OUTPUTS ERROR CODE TO SYSTEM TELETYPE
0083*      CALLING SEQUENCE:
0084*      LDA <2 DIGIT ASCII ERROR CODE>
0085*      JSB ERROR
0086 04344 000000 ERROR NOP
0087 04345 072354R STA MESSE+2
0088 04346 060057 LDA ,4
0089 04347 066352R LDB MESSE
0090 04350 016331R JSB SYOUT
0091 04351 126344R JMP ERROR,I
0092 04352 004353R MESSE DEF **1
0093 04353 046040 ASC 2,L
0094*      SPACE SINGLE SPACES THE LIST OUTPUT DEVICE
0095*      CALLING SEQUENCE:
0096*      JSB SPACE
0097*
0098 04355 000000 SPACE NOP
0099 04356 002404 CLA,INA
0100 04357 066362R LDB ANEW
0101 04360 016364R JSB DRKEY
0102 04361 126355R JMP SPACE,I
0103 04362 004363R ANEW DEF **1
0104 04363 020040 ASC 1,
0105*      DRKEY PRINTS A MESSAGE ON THE LIST OUTPUT DEVICE
0106*      CALLING SEQUENCE:
0107*      LDA <# CHARACTERS (POSITIVE)>
0108*      LDB <ADDRESS OF MESSAGE>
0109*      JSB DRKEY
0110 04364 000000 DRKEY NOP
0111 04365 003004 CMA,INA
0112 04366 073444R STA TEMP
0113 04367 076374R STB **5
0114 04370 016001X JSB EXEC WRITE ON LIST DEVICE
0115 04371 004376R DEF **5
0116 04372 000055 DEF .2
0117 04373 011463R DEF LISTU
0118 04374 004374R DEF *
0119 04375 011444R DEF TEMP
0120 04376 126364R JMP DRKEY,I

```

```

0122* DDOUT WRITES THE CONTENTS OF DBUF ON THE DISC AT THE CURRENT
0123* DISC SECTOR. FOLLOWING THIS, DBUF IS CLEARED, AND THE CURRENT
0124* ADDRESS AND COUNT FOR DBUF ARE SET AND THE NEXT DISC ADDRESS IS
0125* SET INTO DSKAD
0126 04377 000000 DDOUT NOP
0127*
0128 04400 062774R LDA AABUF IF ABFLG=0, OUTPUT IS FROM
0129 04401 066751R LDB ABFLG BUFFER A. SET ABFLG # 0 SO
0130 04402 006002 SZB NEXT OUTPUT WILL BE FROM
0131 04403 062775R LDA ABBUF BUFFER B. SET CURAD SO
0132 04404 072442R STA EBUF BUFFER B WILL BE FILLED NEXT
0133 04405 006002 SZB IF ABFLG#0, OUTUT IS FROM
0134 04406 002400 CLA BUFFER B. SET ABFLG = 0 SO
0135 04407 072751R STA ABFLG NEXT OUTPUT WILL BE FROM
0136 04410 062775R LDA ABBUF BUFFER A. SET CURAD SO
0137 04411 006002 SZB BUFFER A WILL BE FILLED
0138 04412 062774R LDA AABUF NEXT.
0139 04413 073336R STA CURAD
0140*
0141 04414 063354R LDA DSKAD BREAK DOWN DISC ADDR INTO TRACK
0142 04415 017430R JSB GTS SECTOR FOR EXEC WRITE
0143*
0144 04416 063354R LDA DSKAD INCREMENT DISC ADDRESS
0145 04417 016451R JSB DISKA FOR NEXT TIME
0146 04420 073354R STA DSKAD
0147*
0148 04421 063547R LDA N128 INITIALIZE BUFFER COUNT
0149 04422 073350R STA DCNT TO -128
0150*
0151 04423 062450R LDA FIRST IS THIS THE 1ST TIME THROUGH?
0152 04424 002002 SZA
0153 04425 026432R JMP NFRST
0154 04426 060055 LDA ,2 YES = SET DISC LU = 2 FOR
0155 04427 072746R STA DLU SYSTEM DISC
0156 04430 072450R STA FIRST SETFOR NOT 1ST TIME
0157 04431 026434R JMP CL BYPASS WAIT
0158*
0159 04432 017413R NFRST JSB WAIT WAIT FOR CURRENT OPERATION.
0160 04433 037367R ISZ DSCNT INCREMENT # SECTORS OUTPUT.
0161*
0162 04434 067336R CL LDB CURAD CLEAR NEXT BUFFER TO BE
0163 04435 016516R JSB BUFCL FILLED
0164*
0165 04436 016001X JSB EXEC WRITE RELOCATABLE RECORD
0166 04437 004446R DEF ++7
0167 04440 000051 DEF N2 ONTO SYSTEM SCRATCH
0168 04441 004447R DEF WCMND AREA = WITHOUT WAIT
0169 04442 000000 EBUF NOP (FILLED FROM ABOVE)
0170 04443 007550R DEF ,128
0171 04444 005442R DEF TR
0172 04445 005441R DEF SECT
0173*
0174 04446 126377R JMP DDOUT,I
0175*
0176 04447 020002 WCMND OCT 20002 WRITE WITHOUT AIT COMMAND
0177 04450 000000 FIRST NOP

```

```

0178*   DISKA INCREMENTS THE CURRENT DISC ADDRESS TO
0179*   PROVIDE THE ADDRESS OF THE SUCCEEDING SECTOR.
0180*   CALLING SEQUENCE:
0181*       LDA <CURRENT DISC ADDRESS>
0182*       JSB DISKA
0183*   RETURN:
0184*       A CONTAINS NEXT DISC ADDRESS
0185   04451 000000 DISKA NOP
0186   04452 002004      INA
0187   04453 070001      STA B
0188   04454 010074      AND M377
0189   04455 050116      CPA SECTR      CHECK IF LAST SECTOR ON TRACK
0190   04456 026461R     JMP **+3      INCR TRACK #
0191   04457 060001      LDA B
0192   04460 126451R     JMP DISKA,I    RETURN
0193   04461 060001      LDA B
0194   04462 010075      AND M1774
0195   04463 043544R     ADA M400      INCR TRACK #
0196   04464 126451R     JMP DISKA,I    RETURN
0197*
0198*   DISKI READS A SECTOR OF DATA INTO CORE FROM DISC
0199*   CALLING SEQUENCE:
0200*       LDA <DISC T/S ADDRESS>
0201*       LDB <CORE ADDRESS>
0202*       JSB DISKI
0203*
0204   04465 000000 DISKI NOP
0205   04466 076507R     STB COREA      SAVE CORE BUFF ADDR IN EXEC CALL
0206   04467 017430R     JSB GTS        GET TRACK/SECTOR OF DISC ADDR
0207   04470 003400      CCA            SET FOR READ
0208   04471 016501R     JSB I.O        PERFORM I/O
0209   04472 126465R     JMP DISKI,I    RETURN
0210*
0211*   DISKO WRITES A SECTOR OF DATA ONTO DISC FROM CORE
0212*   CALLING SEQUENCE:
0213*       LDA <DISC ADDRESS>
0214*       LDB <CORE ADDRESS>
0215*       JSB DISKO
0216*
0217   04473 000000 DISKO NOP
0218   04474 076507R     STB COREA      SAVE CORE BUFF ADDR IN EXEC CALL
0219   04475 017430R     JSB GTS        GET TRACK/SECTOR OF DISC ADDR
0220   04476 0600051     LDA N2        SET FOR WRITE
0221   04477 016501R     JSB I.O        PERFORM I.O
0222   04500 126473R     JMP DISKO,I    RETURN
0223*
0224   04501 000000 I.O  NOP
0225   04502 072514R     STA RQ        SAVE REQUEST CODE
0226*
0227   04503 016001X     JSB EXEC      EXEC CALL FOR DISC I/O
0228   04504 004513R     DEF **+7
0229   04505 004514R     DEF RQ
0230   04506 011352R     DEF DISCX
0231   04507 000000 COREA NOP
0232   04510 004515R     DEF #WRDS
0233   04511 005442R     DEF TR

```

```

0234 04512 005441R      DEF SECT
0235 04513 126501R      JMP I,0,I
0236*
0237 04514 000000      RQ      NOP      REQUEST CODE"READ"=1, WRITE"=2
0238 04515 000200      #WRDS DEC 128      MAY BE CHANGED BY BLOCK ROUTINE
0239*
0240*      BUFCL CLEARS A 128 WORD BUFFER WITH ZEROS
0241*      CALLING SEQUENCE:
0242*      LDB <BUFFER ADDRESS>
0243*      JSB BUFCL
0244*
0245 04516 000000      BUFCL NOP
0246 04517 063547R      LDA N128
0247 04520 073454R      STA WDCNT      SET CLEAR WORD COUNT
0248 04521 002400      CLA
0249 04522 170001      STA B,I      CLEAR 1 WORD OF BUFFER
0250 04523 006004      INB
0251 04524 037454R      ISZ WDCNT      ALL WORDS CLEARED
0252 04525 026522R      JMP *-3      NO, SO GO CLEAR NEXT WORD
0253 04526 126516R      JMP BUFCL,I      YES, SO RETURN
0254*
0255*      CLID3 CLEARS USAGE FLAGS TO INSURE THAT PROGRAMS WILL BE
0256*      RELOADED AGAIN IF CALLED MORE THAN ONE. THIS IS ESSENTIAL FOR
0257*      ALL LIBRARY AND USER SUBROUTINES, USER MAIN PROGRAMS HOWEVER
0258*      MUST NOT BE CLEARED, BOTH THE USAGE FLAG IN THE IDENT ENTRY
0259*      AND THE BP LINK ADDRESS FOR ALL ENTRY POINTS IN THE PROGRAM ARE
0260*      CLEARED.
0261*
0262 04527 000000      CLID3 NOP
0263 04530 000000      LDA .5
0264 04531 067564R      LDB PLAD      GET ADDRESS OF PROGRAM LIST
0265 04532 053425R      CPA PTYPE      PROG = SEGMENT?
0266 04533 067422R      LDB PLMAN      YES=GET MAIN PROG LIST ADDRESS
0267 04534 077341R      STB CURAP      SET CURRENT PROG LIST ADDRESS
0268 04535 007004      CMB,INB
0269 04536 047564R      ADB PLAD      ADD PROGRAM LIST ADDRESS
0270 04537 047550R      ADB .128      ADJUST FOR PROG LIST LENGTH
0271 04540 007000      CMB
0272 04541 077417R      STB PLCNT      SET CURRENT PROG LIST COUNT
0273 04542 163341R      TRID3 LDA CURAP,I      GET CURRENT PROG LIST ENTRY
0274 04543 002003      SZA,RSS      SKIP=NOT ENT OF PROGLIST
0275 04544 126527R      JMP CLID3,I      RETURN=DONE
0276 04545 037341R      ISZ CURAP      INCR CURRENT PROG LIST ADDR
0277 04546 037417R      ISZ PLCNT      SKIP=END OF PROG LIST
0278 04547 002001      RSS
0279 04550 126527R      JMP CLID3,I      RETURN=DONE
0280 04551 073306R      STA TIDNT      SET CURRENT IDENT ADDR
0281 04552 016266R      JSB IDX      SET IDENT ADDRESSES
0282 04553 026657R      JMP RCERR
0283 04554 163316R      LDA ID6,I      GET M/S , TYPE
0284 04555 002021      SSA,RSS
0285 04556 026565R      JMP TRID5      CLEAR IF NO TRANSFER ADDRESS
0286 04557 010073      AND M177      IF TRANSFER ADDRESS, CHECK
0287 04558 050051      CPA .6      TO SEE IF TYPE INDICATES
0288 04561 002001      RSS      SUBROUTINE (0 OR 1)
0289 04562 050062      CPA .7      YES= CLEAR ID3

```

```

0290 04563 002001      RSS
0291 04564 026542R    JMP TRID3          NO - IGNORE
0292 04565 163313R    TR105 LDA ID3,I
0293 04566 010075      AND M1774
0294 04567 173313R    STA ID3,I
0295 04570 016230R    JSB INLST      INITIALIZE LSTX
0296 04571 016234R    CLSUT JSB LSTX  SET CURRENT LST ADDRESSES
0297 04572 026542R    JMP TRID3      TRY NEXT IDENT
0298 04573 163303R    LDA LST4,I     SET IDENT ADDRESS
0299 04574 053311R    CPA ID1        ENT/EXT BELONGS TO CURRENT PROG
0300 04575 006401      CLB,RSS      YES
0301 04576 026571R    JMP CLSUT      TRY NEXT LST ENTRY
0302 04577 177304R    STB LST5,I     CLEAR BP LINK ADDRESS
0303 04600 026571R    JMP CLSUT      TRY NEXT LST ENTRY
0304*
0305*      BPSCN SCANS LOADER BP AREA FOR A VALUE EQUAL TO THE CURRENT
0306*      OPERAND. IF A VALLUE IS FOUND, THE ADDRESS OF THE OPERAND IS
0307*      RETURNED IN THE A REGISTER. OTHERWISE A NEW BP LINK WORD IS
0308*      RESERVED AND THE ADDRESSSS OF THIS WORD IS RETURNED IN A.
0309*
0310 04601 000000      BPSCN NOP
0311 04602 063456R    LDA UBFWA      LOW BASE PAGE ADDRESS
0312 04603 073405R    STA LOWER      SAVE LOWER BOUND
0313 04604 067441R    LDB TBREL      HIGH BASE PAGE ADDRESS
0314 04605 007004      CMB,INB
0315 04606 077451R    STB UPPER      SAVE NEG UPPER BOUND
0316 04607 063405R    BPSRC LDA LOWER GET CURRENT LOW BP BOUND
0317 04610 043451R    ADA UPPER      TEST FOR END OF AREA
0318 04611 002021      SSA,RSS      NOT END OF AREA
0319 04612 026625R    JMP BPALC      GO TO ALLOCATE LOCATION
0320 04613 067411R    LDB OPRND      GET OPERAND
0321 04614 063405R    LDA LOWER
0322 04615 043331R    ADA BPOIT
0323 04616 043511R    ADA BPBUF      ADJUST FOR START OF BP BUFFER
0324 04617 037405R    ISZ LOWER      INCR LOWER BOUND
0325 04620 154000      CPB A,I       CHECK IF OPERANDS SAME
0326 04621 002001      RSS
0327 04622 026607R    JMP BPSRC      NO-TRY NEXT VALUE
0328 04623 043407R    ADA NBPBF      READJUST FOR BP BUFFER ADDR
0329 04624 126601R    JMP BPSCN,I    RETURN WITH ADDR IN A
0330 04625 017252R    BPALC JSB DLINK GO CHECK FOR CURRENT PAGE LINK
0331 04626 016702R    JSB ALLOC      ALLOCATE NEW BP LINK ADDRESS
0332 04627 126601R    JMP BPSCN,I    RETURN WITH ADDR IN A
0333*
0334*      LSTEX LISTS THE UNDEFINED EXTERNAL REFERENCES
0335*      CALLING SEQUENCE:
0336*      JSB LSTEX
0337*
0338 04630 000000      LSTEX NOP
0339 04631 002400      CLA
0340 04632 073371R    STA EXCNT      CLEAR UNDEFINED EXT COUNT
0341 04633 016230R    JSB INLST
0342 04634 016234R    LST10 JSB LSTX
0343 04635 026654R    JMP LST20      DONE LISTING UNDEFINED EXT
0344 04636 163303R    LDA LST4,I
0345 04637 002021      SSA,RSS

```



```

0346 04640 026634R      JMP LST10          NOT UNDEFINED, SO GET NEXT
0347 04641 063371R      LDA EXCNT          CHECK IF UNDEFINED EXT HEADING
0348 04642 002002      SZA                PUT OUT
0349 04643 026647R      JMP **4          YES
0350 04644 063434R      LDA ,14          NO
0351 04645 066661R      LDB MESUX
0352 04646 016331R      JSB SYOUT          OUTPUT HEADING
0353 04647 060060      LDA ,5
0354 04650 067300R      LDB LST1
0355 04651 016331R      JSB SYOUT
0356 04652 037371R      ISZ EXCNT          INCR UNDEFINED EXT COUNT.
0357 04653 026634R      JMP LST10          GET NEXT LST
0358 04654 063371R LST20 LDA EXCNT          CHECK IF ANY UNDEFINED EXTERNALS
0359 04655 002002      SZA                SKIP=NO
0360 04656 126630R      JMP LSTEX,I      RETURN TO P+1 -UNDEFINED EXTERNAL
0361 04657 036630R      ISZ LSTEX          INCR RETURN ADDRESS
0362 04660 126630R      JMP LSTEX,I      RETURN TO P+2
0363 04661 004662R MESUX DEF **1
0364 04662 052516      ASC 7,UNDEFINED EXTS
0365*
0366*      SETOP SETS CURRENT OPERAND IN TH BP LOCATION SPECIFIED
0367* BY TH ADDRESS IN A
0368*
0369 04671 000000      SETOP NOP
0370 04672 072701R      STA BPADD
0371 04673 043331R      ADA BPOIT
0372 04674 043511R      ADA BPBUF          ADJUST BP LINK ADDRESS
0373 04675 067411R      LDB OPRND          GET AND STORE OPERAND
0374 04676 174000      STB A,I          INTO BP LOCATION
0375 04677 062701R      LDA BPADD
0376 04700 126671R      JMP SETOP,I      RETURN
0377 04701 000000      BPADD NOP
0378*
0379*      ALLOC ESTABLISHES ALL THE BASE PAGE LINKAGE ADDRESSES.
0380* DIAGNOSTIC IS OUTPUT IF OUT OF SPACE.
0381*
0382 04702 000000      ALLOC NOP
0383 04703 063441R      LDA TBREL          GET NEXT BP ADDRESS
0384 04704 043331R      ADA BPOIT
0385 04705 043511R      ADA BPBUF          ADJUST BP ADDRESS
0386 04706 006400      CLB
0387 04707 174000      STB A,I          CLEAR ALLOCATED BP LOCATION
0388 04710 063441R      LDA TBREL
0389 04711 037441R      ISZ TBREL          INCR NEXT BP ADDRESS
0390 04712 064000      LDB A          CHECK IF BASE PAGE OVERFLOW
0391 04713 007000      CMB
0392 04714 047457R      ADB UBLWA
0393 04715 006021      SSB,RSS
0394 04716 126702R      JMP ALLOC,I
0395 04717 063472R      LDA ERR04          BP LINK OVERFLOW ERROR
0396 04720 026631R      JMP ABRTX          ABORT LOADR
0397*
0398*      DBSET GIVES THE ADDRESS OF THE NEXT WORD OF THE RELOCATABLE
0399* RECORD IN LBUF, IF AT END OF BUFFER, IT READS IN ANOTHER PACKED
0400* RELOCATABLE RECORD
0401*

```

PAGE 0069 #05 DOS-III LOADER- UTILITY ROUTINES

0402	04721	000000	DBSET	NOP	
0403	04722	037340R	ISZ	CURAL	INCR CURRENT LBUF ADDRESS
0404	04723	037403R	ISZ	LCNT	SKIP-END OF LBUF
0405	04724	126721R	JMP	DBSET,I	RETURN
0406	04725	017376R	JSB	INREL	READ NEXT RELOCAABLE RECORD
0407	04726	126721R	JMP	DBSET,I	RETURN

```

0409*
0410* INPUT NAM RECORD FROM DISC
0411*
0412 04727 000000 INNAM NOP
0413 04730 073361R STA DSKRD SAVE ORIGINAL DISC ADDRESS
0414*
0415 04731 063352R LDA DISCX
0416 04732 072746R STA DLU DEPENDING ON DISCX
0417*
0418 04733 032747R IOR WOWC SET UP READ COMMAND
0419 04734 072750R STA CMND WITHOUT WAIT
0420*
0421 04735 017413R JSB WAIT CHECK IF DISC IS BUSY
0422 04736 002400 CLA CLEAR # SECTORS
0423 04737 073367R STA DSCNT READ
0424 04740 072751R STA ABFLG SET FOR BUFFER A
0425*
0426 04741 063361R LDA DSKRD BREAK DOWN DISC ADDRESS FOR
0427 04742 017430R JSB GTS READ
0428*
0429 04743 016752R JSB DRWOW ISSUE READ WITHOUT WAIT FOR BU A
0430*
0431 04744 017376R JSB INREL SET FOR B, WAIT FOR A, REQUEST B
0432*
0433 04745 126727R JMP INNAM,I
0434*
0435 04746 000000 DLU NOP DISC LOGICAL UNIT #
0436 04747 020000 WOWC OCT 20000 WITHOUT WAIT
0437 04750 000000 CMND NOP
0438 04751 000000 ABFLG NOP A/B BUFFER FLAG

```

```

0440*
0441* DISK READ WITHOUT WAIT
0442*
0443 04752 000000 DRWOW NOP
0444*
0445 04753 062774R LDA AABUF IF ABFLG#0, SET FOR INPUT
0446 04754 066751R LDB ABFLG INTO AABUF THEN SET
0447 04755 006002 SZB ABFLG#0.
0448 04756 062775R LDA ABBUF IF ABFLG#0, SET FOR INPUT
0449 04757 072767R STA CBUF INTO BBUF, THEN SET
0450 04760 006002 SZB ABFLG#0.
0451 04761 002400 CLA
0452 04762 072751R STA ABFLG
0453*
0454 04763 016001X JSB EXEC
0455 04764 004773R DEF **7
0456 04765 000052 DEF N1
0457 04766 004750R DEF CMND
0458 04767 000000 CBUF NOP SET FROM ABFLG
0459 04770 007550R DEF ,120
0460 04771 005442R DEF TR
0461 04772 005441R DEF SECT
0462*

```

PAGE 0071 #05 DOS-III LOADER- UTILITY ROUTINES

0463 04773 126752R JMP DRWOW,I

0464*

0465 04774 004776R AABUF DEF ABUF

0466 04775 005176R ABBUF DEF BBUF

0467 04776 000000 ABUF BSS 128

0468 05176 000000 BBUF BSS 128

0470*

0471* INPUT RELOCATABLE RECORD

0472*

0473 05376 000000 INREL NOP

0474*

0475 05377 063361R LDA DSKRD

INCREMENT DISC

0476 05400 017443R JSB NDA

COMPUTE NEXT DISC ADDRESS

0477 05401 073361R STA DSKRD

BREAK INTO TRACK/SECTOR

0478 05402 017430R JSB GTS

0479*

0480 05403 017413R JSB WAIT

0481*

0482 05404 062767R LDA CBUF

INITIALIZE RELOCATABLE INPUT
BUFFER POINTERS

0483 05405 073340R STA CURAL

0484 05406 063547R LDA N128

0485 05407 073403R STA LCNT

0486*

0487 05410 037367R ISZ DSCNT

INCREMENT # SECTOR READ

0488*

0489 05411 016752R JSB DRWOW

INITIATE NEXT READ

0490*

0491 05412 127376R JMP INREL,I

0493*

0494* WAIT FOR CURRENT DISC TRANSFER TO COMPLTE

0495*

0496 05413 000000 WAIT NOP

0497*

0498 05414 016001X JSB EXEC

0499 05415 005422R DEF **5

0500 05416 005425R DEF N13

STATUS REQUEST

0501 05417 004746R DEF DLU

LOGICAL UNIT SPECIFIED AT NAM

0502 05420 005426R DEF STAT

0503 05421 005427R DEF TLOG

0504*

0505 05422 002020 SSA

IF BUSY,

0506 05423 027414R JMP WAIT+1

LOOP

0507*

0508 05424 127413R JMP WAIT,I

0509*

0510 05425 177763 N13 DEC -13

0511 05426 000000 STAT NOP

0512 05427 000000 TLOG NOP

```

0514*
0515* GET TRACK/SECTOR FROM CURRENT DISC ADDRESS
0516*
0517 05430 000000 GTS NOP
0518*
0519 05431 070001 STA B SAVE CURRENT DISC ADDRESS
0520 05432 010074 AND M377
0521 05433 073441R STA SECT SECTOR # IN LOW 8 BITS
0522 05434 000001 LDA B
0523 05435 001727 ALF,ALF
0524 05436 010074 AND M377
0525 05437 073442R STA TR TRACK # IN UPPER 8 BITS
0526*
0527 05440 127430R JMP GTS,I
0528*
0529 05441 000000 SECT NOP CURRENT DISC SECTOR #
0530 05442 000000 TR NOP CURRENT DISC TRACK #

0532*
0533* NEXT DISC ADDRESS
0534*
0535 05443 000000 NDA NOP
0536*
0537 05444 010451R JSB DISKA ADDRESS NORMALLY
0538*
0539 05445 007443R LDB TDIR IF FORWARD INCREMNT,
0540 05446 006003 SZB,RSS NEW DISC ADDRESS IS
0541 05447 127443R JMP NDA,I IN "A"
0542*
0543 05450 070001 STA B IF BACKWARD INCREMENT,
0544 05451 010074 AND M377 FIRST CHECK TO SEE
0545 05452 002002 SZA IF NEW TRACK
0546 05453 027457R JMP NDA2 NO- VALUE IN B IS OK
0547*
0548 05454 005727 BLF,BLF YES- DECREMENT BACK 2
0549 05455 044051 ADB N2 TRACKS
0550 05456 005727 BLF,BLF ROTATE TO HIGH 8 BITS
0551*
0552 05457 060001 NDA2 LDA B UPDATE NW
0553*
0554 05460 127443R JMP NDA,I

```

```

0556*
0557*      GENID GENERATEES THE DIRECTORY ENTRY FOR THE PROGRAM LOADED.
0558*
0559 05461 000000 GENID NOP
0560 05462 003400 CCA
0561 05463 073420R STA PLFLG
0562 05464 063400R LDA IMAIN      SET ADDRESS FOR IDX
0563 05465 073306R STA TIDNT
0564 05466 016265R JSB IDX
0565 05467 026637R JMP RCERR
0566 05470 163311R LDA ID1,I
0567 05471 017662R JSB OUTID      NAME 1,2 TO DIRECTORY
0568 05472 163312R LDA ID2,I
0569 05473 017662R JSB OUTID      NAME 3,4 TO DIRECTORY
0570 05474 163316R LDA ID6,I      GET TYPE
0571 05475 010073 AND M177
0572 05476 067462R LDB DVR        LOADING DRIVERS?
0573 05477 054057 CPB .4
0574 05500 060057 LDA .4          YES - RESET TO TYPE 4
0575 05501 033542R IOR M200       SET -P- BIT
0576 05502 070001 STA B
0577 05503 163313R LDA ID3,I      NAME 5
0578 05504 010075 AND M1774
0579 05505 030001 IOR B
0580 05506 017662R JSB OUTID      NAME 5,P,TYPE TO DIRECTORY
0581 05507 063360R LDA DSKMN
0582 05510 017662R JSB OUTID      MAIN DISC ADDR TO DIRECTORY
0583 05511 063423R LDA PPREL
0584 05512 067425R LDB PTYPE
0585 05513 054060 CPB .5
0586 05514 063325R LDA BSPAD
0587 05515 043326R ADA BSSDP
0588 05516 072210R STA TBUF
0589 05517 003004 CMA,INA
0590 05520 043603R ADA TPREL
0591 05521 043440R ADA .127
0592 05822 013545R AND M7600
0593 05523 001727 ALF,ALF
0594 05524 001200 RAL
0595 05525 072211R STA TBUF+1     MAIN # OF SECTORS
0596 05526 063413R LDA PBREL
0597 05527 054060 CPB .5
0598 05530 063324R LDA BSBAD
0599 05531 072212R STA TBUF+2     LOW BASE
0600 05532 003004 CMA,INA
0601 05533 043441R ADA TBREL
0602 05534 043440R ADA .127
0603 05535 013545R AND M7600
0604 05536 001727 ALF,ALF
0605 05537 001200 RAL
0606 05540 042211R ADA TBUF+1     TOTAL # OF SECTORS
0607 05541 017662R JSB OUTID      NUMBER OF SECTORS TO DIRECTORY
0608 05542 062210R LDA TBUF
0609 05543 017662R JSB OUTID      LOW MAIN ADDR TO DIRECTORY
0610 05544 063603R LDA TPREL
0611 05545 017662R JSB OUTID      HIGH MAIN ADDR TO DIRECTORY

```

0612	05546	062212R	LDA TBUF+2	
0613	05547	017662R	JSB OUTID	LOW BASE ADDRESS TO DIRECTORY
0614	05550	063441R	LDA TBREL	
0615	05551	017662R	JSB OUTID	HIGH BASE ADDR TO DIRECTORY
0616*				
0617	05552	067462R	LDB DVR	IF LOADING I/O DRIVERS, FIND
0618	05553	054057	CPB .4	I.XX AND C.XX ENTRY POINTS
0619	05554	027604R	JMP DVRAD	FOR DIRECTORY ENTRY
0620*				
0621	05555	063424R	LDA PRENT	
0622	05556	017662R	JSB OUTID	PRIMARY ENTRY ADDRESS TO DIRECT
0623	05557	063603R	LDA TPREL	
0624	05560	017662R	JSB OUTID	LAST WORD ADDRESS TO DIRECTORY
0625	05561	067375R	DVRTN LDB ICNT	CHECK IF BEGINNING OF NEW SECTOR
0626	05562	057547R	CPB N128	
0627	05563	127461R	JMP GENID,I	NEW SECTOR BEGUN RETURN
0628	05564	063375R	LDA ICNT	CHECK IF ROOM FOR ANOTHER
0629	05565	040065	ADA .10	DIRECTORY ENTRY
0630	05566	002020	SSA	
0631	05567	127461R	JMP GENID,I	ROOM FOR ANOTHER ENTRY RETURN
0632	05570	003400	CCA	
0633	05571	017662R	JSB OUTID	SET ANOTHER DIRECTORY ENTRY. FOLLO
0634	05572	067375R	LDB ICNT	
0635	05573	077444R	STB TEMP	SET ZERO OUTPUT COUNT
0636	05574	057547R	CPB N128	
0637	05575	127461R	JMP GENID,I	
0638	05576	002400	CLA	
0639	05577	017662R	JSB OUTID	OUTPUT ZERO TO DIR BUFFER
0640	05600	037444R	ISZ TEMP	
0641	05601	027576R	JMP *-3	OUTPUT ANOTHER ZERO TO BUFFER
0642	05602	127461R	JMP GENID,I	RETURN
0643	05603	000000	LIBRY NQP	FIRST LIBR PROG LOAD ADDRESS
0644*				
0645*	DVRAD			SEARCHES THE LST FOR THE I.XX & C.XX
0646*				ENTRY POINTS FOR THE I/O DRIVER BEING
0647*				LOADED.
0648*				THESE VALUES ARE THEN STORED IN
0649*				WORDS 10 & 11 OF THE DIRECTORY ENTRY
0650*				
0651	05604	163312R	DVRAD LDA ID2,I	FIND THE XX NUMBER FROM
0652	05605	010073	AND M177	THE DVRXX NAME. STORE
0653	05606	001727	ALF,ALF	THE ASCII XX IN TEMP.
0654	05607	073444R	STA TEMP	
0655	05610	163313R	LDA ID3,I	
0656	05611	001727	ALF,ALF	
0657	05612	010073	AND M177	
0658	05613	033444R	IOR TEMP	
0659	05614	073444R	STA TEMP	
0660*				
0661	05615	016230R	JSB INLST	SEARCH LST
0662	05616	016234R	DVR1 JSB LSTX	
0663	05617	027634R	JMP DVR2	
0664	05620	163300R	LDA LST1,I	
0665	05621	053660R	CPA IDEC	I. ?
0666	05622	027631R	JMP IP	
0667	05623	053661R	CPA CDEC	C. ?

```

0668 05624 002001      RSS
0669 05625 027616R     JMP DVR1
0670 05626 017641R     JSB DVRS      CHECK FOR CORRECT XX
0671 05627 073603R     STA LIBRY     SET C,XX ADDRESS
0672 05630 027616R     JMP DVR1
0673*
0674 05631 017641R IP   JSB DVRS      CHECK FOR CORRECT XX
0675 05632 073424R     STA PRENT     SET I,XX ADDRESS
0676 05633 027616R     JMP DVR1
0677*
0678 05634 063424R DVR2 LDA PRENT
0679 05635 017662R     JSB OUTID     OUTPUT WORD 10 = I,XX
0680 05636 063603R     LDA LIBRY
0681 05637 017662R     JSB OUTID     OUTPUT WORD 11 = C,XX
0682 05640 027561R     JMP DVRTN
0683*
0684*
0685 05641 000000      DVRS NOP
0686 05642 163301R     LDA LST2,I    XX MATCH?
0687 05643 053444R     CPA TEMP
0688 05644 002001      RSS
0689 05645 027616R     JMP DVR1      NO
0690 05646 163302R     LDA LST3,I
0691 05647 010075      AND M1774
0692 05650 053454R     CPA UBLNK     5TH CAHR = BLANK?
0693 05651 002001      RSS
0694 05652 027616R     JMP DVR1      NO
0695 05653 163304R     LDA LST5,I    GET BP LINK
0696 05654 043331R     ADA BPOIT     COMPUTE ABSOLUTE ADDRESS
0697 05655 043511R     ADA BPBUF
0698 05656 160000      LDA A,I
0699 05657 127641R     JMP DVRS,I    RETURN WITH ADDRESS IN "A"
0700*
0701 05660 044456      IDEC ASC 1,I,
0702 05661 041456      CDEC ASC 1,C,
0703*
0704*
0705*      OUTID STORES DIRECTORY ENTRIES IN ID SEGMENT BUFFER
0706*      AND WRITES IT ON DISC WHEN IT BECOMES FULL.
0707*
0708 05662 000000      OUTID NOP
0709 05663 173337R     STA CURAI,I   STORE WORD INTO ID BUFFER
0710 05664 037337R     ISZ CURAI     INCR BUFFER ADDR
0711 05665 037375R     ISZ ICNT      INCR COUNT
0712 05666 127662R     JMP OUTID,I   RETURN=NOT FULL
0713 05667 016000R     JSB SETCR     SET USER DISC FLAG IN BASE PAGE
0714 05670 063357R     LDA DSKID
0715 05671 067552R     LDB AIBUF
0716 05672 016473R     JSB DISKO     WRITE 1 SECTOR OF DIRECTORY
0717 05673 067552R     LDB AIBUF
0718 05674 077337R     STB CURAI     SET NEW CURRENT IBUF ADDR
0719 05675 016516R     JSB BUFCL     CLEAR BUFFER
0720 05676 060174      LDA TSONE     CURRENT DISC ADDRESS
0721 05677 073357R     STA DSKID
0722 05700 010074      AND M377
0723 05701 002003      SZA,RSS      NO

```


PAGE 0076 #05 D08-III LOADER- UTILITY ROUTINES

0724	05702	026330R	JMP DRCHA	OUT OF DIRECTORY SPACE
0725	05703	063547R	LDA N128	
0726	05704	073375R	STA ICNT	SET CURRENT IBUF COUNT
0727	05705	127662R	JMP OUTID,I	RETURN

```

0729* LABDO PUTS OUT CURRENT ABSOLUTE CODE WORD FOR THE PROGRAM
0730* BEING LOADED.
0731* IF WORD IS IN CURRENT SECTOR, IT IS INSERTED.
0732* IF WORD DOES NOT FIT IN CURRENT SECTOR, THE CURRENT SECTOR
0733* IS OUTPUT AND THE NEW SECTOR READ IN.
0734* ADDITIONALLY, THERE ARE CHECKS FOR MEMORY OVERFLOW AND MEMORY
0735* OVERLAP
0736* CALLING SEQUENCE:
0737* LDA <CURRENT ABSOLUTE CODE WORD>
0738* LDB <CURRENT ABSOLUTE ADDRESS>
0739* JSB LABDO
0740 05706 000000 LABDO NOP
0741 05707 073437R STA TABS ABSOLUTE CODE WORD
0742 05710 077440R STB TADR ABSOLUTE CORE ADDRESS
0743 05711 077427R STB RELAD UPDATE RELOCATION ADDRESS
0744 05712 007004 CMB,INB
0745 05713 047461R ADB UMLWA CHECK IF MEMORY OVERFLOW
0746 05714 006021 SSB,RSS
0747 05715 027720R JMP LAB10 NO
0748 05716 063471R LDA ERR03 YES-NO OUTPUT MESSAGE
0749 05717 026631R JMP ABRTX AND ABORT
0750 05720 063332R LAB10 LDA CBASE COMPUTE # OF WORDS
0751 05721 003004 CMA,INA
0752 05722 043440R ADA TADR
0753 05723 016265R JSB RRL9
0754 05724 073450R STA TWORD WORD #
0755 05725 007004 CMB,INB
0756 05726 077442R STB TCNT
0757 05727 063343R LDA DBASE
0758 05730 006003 SZB,RSS
0759 05731 027735R JMP ++4
0760 05732 016451R JSB DISKA
0761 05733 037442R ISZ TCNT
0762 05734 027732R JMP *-2
0763 05735 073446R STA TNEXT ABSOLUTE TRACK/SECTOR #
0764 05736 053351R CPA DCRNT CHECK IF CURRENT SECTOR
0765 05737 027744R JMP LAB20 YES
0766 05740 016115R JSB REMDO NO- WRITE OUT CURRENT SECTOR
0767 05741 063446R LDA TNEXT
0768 05742 073351R STA DCRNT
0769 05743 016011R JSB CLRTK SNOWFLOW TRACKS
0770 05744 067551R LAB20 LDB ADBUF COMPUTE ADDRESS TO STORE WORD
0771 05745 047450R ADB TWORD
0772 05746 063437R LDA TABS CHECK IF ABS WORD IS HALT 0
0773 05747 002003 SZA,RSS
0774 05750 027770R JMP LAB40 YES-NO DON'T STORE
0775 05751 160001 LDA B,I SO-NO CHECK IF WORD IN BUFFER
0776 05752 002003 SZA,RSS IS 0
0001 05753 027766R JMP LAB30 YES-NO GO STORE
0002 05754 063347R LDA DBUGF CHECK IF DEBUG FLAG SET
0003 05755 002002 SZA NO-NO MEMORY OVERLAY
0004 05756 027764R JMP LAB25 YES
0005 05757 063421R LDA PLGTH CHECK IF COMPILER PRODUCED PROG
0006 05760 002020 SSA SKIP-ASSEMBLER PRODUCED
0007 05761 027766R JMP LAB30
0008 05762 063506R LDA ERR16 NO-NO PROGRAM OVERLAY

```

```

0009 05763 026631R      JMP ABRTX
0010 05764 002400 LAB25 CLA
0011 05765 073347R-    STA DBUGF      CLEAR DEBUG FLAG
0012 05766 063437R LAB30 LDA TABS
0013 05767 170001      STA B,I      STORE WORD IN BURFER
0014 05770 063427R LAB40 LDA RELAD    CHECK IF NEW MAXIMUM ADDRESS
0015 05771 070001      STA B
0016 05772 003004      CMA,INA
0017 05773 043406R     ADA MAXAD
0018 05774 002021      SSA,RSS
0019 05775 127706R     JMP LABDO,I   RETURN
0020 05776 077406R     STB MAXAD    SET NEW MAX ADDRESS
0021 05777 127706R     JMP LABDO,I   RETURN

```

PAGE 0079 #06 DOS=III LOADER- UTILITY ROUTINES

0023	06000	000000	SETCR NOP	SET USER DISC FLAG IN BASE PAGE
0024	06001	060056	LDA ,3	
0025	06002	073352R	STA DISCX	
0026	06003	126000R	JMP SETCR,I	
0027*				
0028*	BASE PAGE STORE			
0029*				
0030	06004	000000	BPSTR NOP	STORE A IN B
0031	06005	016001X	JSB EXEC	
0032	06006	006010R	DEF ++2	
0033	06007	007430R	DEF N19	
0034	06010	126004R	JMP BPSTR,I	
0035*				
0036*	CLEAR TRACKS			
0037*				
0038	06011	000000	CLRTK NOP	
0039	06012	001727	ALF,ALF	RIGHT JUSTIFY
0040	06013	010074	AND M377	TO LOW 8 BITS
0041	06014	043363R	ADA DSKTM	COMPARE TO UPPR TRACK LIMIT
0042	06015	002021	SSA,RSS	NOT TO BIG IF NEG
0043	06016	026330R	JMP DRCHA	ERROR L10
0044	06017	063351R	LDA DCRNT	CURRENT REQUEST T/S

```

0079 06062 010074      AND M377
0080 06063 003004      CMA,INA
0081 06064 073365R     STA DSK02      COMP OF LAST SECTOR CLEARED
0082 06065 016000R LAB19 JSB SETCR
0083 06066 063361R     LDA DCRNT
0084 06067 067551R     LDB ADBUF
0085 06070 016465R     JSB DISKI      READ IN DESIRED SECTOR
0086 06071 126011R     JMP CLRTK,I
0087*
0088* INITIALIZE 1ST DISC TRACK
0089*
0090 06072 000000      INDSK NOP
0091 06073 072115R     STA REMDO      SAVE TEMP
0092 06074 010075      AND M1774      GET START TRAK NUM
0093 06075 001727      ALF,ALF        RIGHT JUSTIFY
0094 06076 003004      CMA,INA        MAKE NEG
0095 06077 073364R     STA DSK01
0096 06100 062115R     LDA REMDO
0097 06101 010074      AND M377      GET START SECTOR NUM
0098 06102 003004      CMA,INA
0099 06103 073365R     STA DSK02
0100 06104 067551R     LDB ADBUF
0101 06105 016516R     JSB BUFCL
0102 06106 016000R     JSB SETCR
0103 06107 062115R     LDA REMDO
0104 06110 067551R     LDB ADBUF      CORA ADDRESS
0105 06111 016473R     JSB DISKO      WRITE DISC
0106 06112 060174      LDA TSONE      NEXT AVAILABLE T/S
0107 06113 073366R     STA DSKUD
0108 06114 126072R     JMP INDSK,I
0109*
0110* REMDO OUTPUTS CURRENT BUFFER.
0111* NORMALLY USED WHEN CODE FOR MAIN PROGRAM OR SEGMENT HAS BEEN
0112* GENERATED.
0113*
0114 06115 000000      REMDO NOP
0115 06116 016000R     JSB SETCR      SET USER DISC FLAG IN BASE PAGE
0116 06117 063351R     LDA DCRNT
0117 06120 067551R     LDB ADBUF
0118 06121 016473R     JSB DISKO
0119 06122 126115R     JMP REMDO,I
0120* READ IN INITIAL DISC SECTOR
0121 06123 000000      DINIT NOP
0122 06124 063354R     LDA DSKAD      SACE INITIAL DISC ADDRESS
0123 06125 073343R     STA DBASE
0124 06126 073351R     STA DCRNT
0125 06127 016072R     JSB INDSK      INITILIZE START DISC TRAK ADD
0126 06130 002400      CLA
0127 06131 073406R     STA MAXAD      CLEAR MAXIMUM ADDRESS FLAG
0128 06132 126123R     JMP DINIT,I
0129* COMPUTE AND UPDATE DSKAD ADDRESS
0130 06133 000000      RESET NOP
0131 06134 062210R     LDA TBUF
0132 06135 072264R     STA TTBUF
0133 06136 006400      CLB
0134 06137 063333R     LDA CLXXX

```

```

0191 06230 072212R STA TBUF+2
0192 06231 037444R ISZ TEMP
0193 06232 017341R JSB LSCAN SEARCH LST FOR SYMBOL
0194 06233 026244R JMP RES70 NOT FOUND
0195 06234 163302R LDA LST3,I FOUND
0196 06235 010075 AND M1774
0197 06236 173302R STA LST3,I CLEAR ORDINAL
0198 06237 163303R LDA LST4,I
0199 06240 001665 ELA,CLE,ERA
0200 06241 173303R STA LST4,I SET IDENT POSITIVE
0201 06242 002400 CLA
0202 06243 173304R STA LST5,I CLEAR BP LINK ADDRESS
0203 06244 037442R RES70 ISZ TCNT
0204 06245 026221R JMP RES60
0205 06246 062264R RES05 LDA TBUF BASE PROGRAM ADDRESS
0206 06247 003004 CMA,INA
0207 06250 043603R ADA TPREL ADD MAXIMUM ADDRESS FLAG
0208 06251 016265R JSB RRL9
0209 06252 002002 SZA
0210 06253 006004 INB
0211 06254 007004 CMB,INB
0212 06255 077442R STB TCNT NEG # SECTORS USED
0213 06256 063343R LDA DBASE GET STARTING DISC ADDRESS
0214 06257 016451R JSB DISKA INCR # OF SECTORS
0215 06260 037442R ISZ TCNT USED TIME
0216 06261 026257R JMP #=2
0217 06262 073354R STA DSKAD STORE NEXT DISC T/S ADDRESS
0218 06263 126133R JMP RESET,I
0219 06264 000000 TBUF NOP
0220 06265 000000 RRL9 NOP DO RRL 9 TO B+A REG
0221 06266 070001 STA B
0222 06267 013545R AND M7600 MASK OF UPPER 9 BITS
0223 06270 001727 ALF,ALF TAKE UPPER 9 BITS TO LOW
0224 06271 001200 RAL
0225 06272 072437R STA CLIST
0226 06273 060001 LDA B
0227 06274 010073 AND M177 MASK LOW 7 BITS
0228 06275 066437R LDB CLIST
0229 06276 126265R JMP RRL9,I
0230* CLRLT CLEARS THE CURRENT BP LINKAGE ADDRESSES IN LST
0231* TO MAKE THEM UNAVAILABLE TO THE SUCCEEDING PROGRAMS. THIS CONSIST
0232* OF CLEARING WORDS OF THE LST ENTRY AND THE CORRESPONDING BP LINK
0233* VALUE IN THE BASE PAGE BUFFER.
0234* CALLING SEQUENCE:
0235* LDA <LOW BASE PAGE ADDRESS>
0236* LDB <HIGH BASE PAGE ADDRESS>
0237* JSB CLRLT
0238 06277 000000 CLRLT NOP
0239 06300 003004 CMA,INA
0240 06301 073405R STA LOWER SAVE LOWER BOUND
0241 06302 007004 CMB,INB
0242 06303 077451R STB UPPER SAVE UPPER BOUND
0243 06304 016230R JSB INLST INITIALIZE LST SCAN
0244 06305 016234R CLRNX JSB LSTX SET LST ADDRESSES
0245 06306 026322R JMP CLBPS GO TO CLEAR BP
0246 06307 163304R LDA LST5,I CHECK IF BP LINK TO BE CLEARED

```

```

0247 06310 043405R ADA LOWER
0248 06311 002020 SSA
0249 06312 026305R JMP CLRNX TRY NEXT ENTRY
0250 06313 163304R LDA LST5,I
0251 06314 043451R ADA UPPER
0252 06315 002021 SSA,RSS
0253 06316 026305R JMP CLRNX TRY NEXT ENTRY
0254 06317 006400 CLB
0255 06320 177304R STB LST5,I CLEAR BP LINK ADDRESS
0256 06321 026305R JMP CLRNX TRY NEXT ENTRY
0257 06322 067405R CLBPS LDB LOWER CLEAR BP LINK BUFFER BETWEEN
0258 06323 007004 CMB,INB UPPRR AND LOWER LIMITS
0259 06324 063451R LDA UPPER
0260 06325 040001 ADA B
0261 06326 002021 SSA,RSS
0262 06327 126277R JMP CLRLT,I RETURN
0263 06330 073454R STA WDCNT
0264 06331 047511R ADB BPBUF ADJUST FOR BASE PAGE
0265 06332 047331R ADB BPOIT
0266 06333 002400 CLA
0267 06334 170001 STA B,I
0268 06335 006004 INB
0269 06336 037454R ISZ WDCNT
0270 06337 026334R JMP *-3
0271 06340 126277R JMP CLRLT,I RETURN
0272* BPOUT OUTPUTS THE BASE PAGE SECTION OF CODE FOLLOWING THE
0273* LOADING OF EACH MAIN OR SEGMENT. DISC ADDRESS IN DSKAD.
0274* CALLING SEQUENCE:
0275* LDA <UPPER BP ADDRESS>
0276* LDB <LOWER BP ADDRESS>
0277* JSB BPOUT
0278 06341 000000 BPOUT NOP
0279 06342 003004 CMA,INA
0280 06343 040001 ADA B
0281 06344 073442R STA TCNT SAVE BP LENGHT
0282 06345 047511R ADB BPBUF ADJUST FOR BP ADDRESS
0283 06346 047331R ADB BPOIT
0284 06347 077342R STB CURAT SAVE CURRENT LOW CORE ADDRESS
0285 06350 002021 SSA,RSS
0286 06351 126341R JMP BPOUT,I RETURN-ALL CODE OUT
0287 06352 016000R BPSYO JSB SETCR USER AREA
0288 06353 063354R LDA DSKAD CURRENT DISC ADDRESS
0289 06354 067342R LDB CURAT CURRENT LOW CORE ADDRESS
0290 06355 016473R JSB DISKO OUTPUT CURRENT BP SECTOR
0291 06356 060174 LDA TSONE CURRENT DISC ADDRESS
0292 06357 073354R STA DSKAD
0293 06360 067442R LDB TCNT
0294 06361 047550R ADB ,128
0295 06362 077442R STB TCNT SAVE REMAINING COUNT
0296 06363 006021 SSB,RSS CHECK IF MORE TO OUTPUT
0297 06364 126341R JMP BPOUT,I RETURN-ALL CODE OUT
0298 06365 067342R LDB CURAT
0299 06366 047550R ADB ,128
0300 06367 077342R STB CURAT SET NEXT CORE ADDRESS
0301 06370 026352R JMP BPSYO OUTPUT NEXT BP SECTOR TO DISC
0302* CONVD CONVERTS CONTENTS OF A TO ASCII DECIMAL OR OCTAL AND

```

```

0303* STORES RESULT AT LOCATION SPECIFIED IN B,
0304* CALLING SEQUENCE:
0305* LDA <# TO BE CONVERTED> +=OCTAL; -=DECIMAL
0306* LDB <ADDR TO STORE RESULT>
0307* JSB CONVD
0308 06371 000000 CONVD NOP
0309 06372 077342R STB CURAT SET STORE ADDRESS
0310 06373 067565R LDB OPWRS ADDRESS OF OCTAL LOWERS
0311 06374 002020 SSA
0312 06375 067573R LDB DPWRS ADDRESS OF DECIMAL POWERS
0313 06376 077426R STB RANAD SET POWER RANGE ADDRESS
0314 06377 002021 SSA,RSS
0315 06400 003004 CMA,INA CONVERT # TO NEGATIVE
0316 06401 070001 STA B
0317 06402 060051 LDA N2
0318 06403 073442R STA TCNT SET CONVERSION COUNTER
0319 06404 016422R JSB GETD
0320 06405 033454R IOR UBLNK
0321 06406 173342R STA CURAT,I SAVE BLANK,FIRST DIGIT
0322 06407 037342R ISZ CURAT
0323 06410 016422R NEXTD JSB GETD GET NEXT DIGIT
0324 06411 001727 ALF,ALF
0325 06412 173342R STA CURAT,I SAVE AS UPPER CHARACTER
0326 06413 016422R JSB GETD GET NEXT DIGIT
0327 06414 133342R IOR CURAT,I ADD UPPER CHARACTER
0328 06415 173342R STA CURAT,I STORE NEXT 2 CHARACTERS
0329 06416 037342R ISZ CURAT
0330 06417 037442R ISZ TCNT
0331 06420 026410R JMP NEXTD GET NEXT 2 DIGITS
0332 06421 126371R JMP CONVD,I RETURN -DONE
0333 06422 000000 GETD NOP
0334 06423 002400 CLA
0335 06424 147426R INCRA ADB RANAD,I ADD POWER
0336 06425 007026 CMB,SSB,INB,SZB
0337 06426 026432R JMP **4 DIGIT FOUND
0338 06427 002004 INA INCR DIGIT
0339 06430 007004 CMB,INB RESTORE REMAINDER TO NEG
0340 06431 026424R JMP INCRA RRY NEXT HIGHER DIGIT
0341 06432 147426R ADB RANAD,I ADD POWER
0342 06433 007004 CMB,INB RESTORE REMAINDER
0343 06434 037426R ISZ RANAD INCR POWER LIST ADDRESS
0344 06435 033541R IOR M60 CONVERT TO ASCII
0345 06436 126422R JMP GETD,I RETURN
0346*
0347* CLIST CLEARS MEMORY MAP BUFFER WITH BLANKS
0348*
0349 06437 000000 CLIST NOP
0350 06440 067560R LDB AMLST ADDRESS OF MLIST
0351 06441 063431R LDA N18
0352 06442 073321R STA AMAD
0353 06443 063452R LDA BLNKS TWO ASCII BLANKS
0354 06444 170001 STA B,I CLEAR BUFFER WORD
0355 06445 006004 INB
0356 06446 037321R ISZ AMAD
0357 06447 026444R JMP **3
0358 06450 126437R JMP CLIST,I RETURN-BUFFER CLEARED

```



```

0359*
0360*      IDSCN SCANS PROGRAM IDENT LIST FOR PROGRAM MAIN OR SEGMENT.
0361*      PROGRAM TYPE SET IN PTYPE.
0362*      CALLING SEQUENCE:
0363*          JSB IDSCN
0364*      RETURN
0365*          (N+1) = END OF IDENTs
0366*          (N+2) = PROG FOUND E=M/S FOR CURRENT PROGRAM.
0367*
0368 06451 000000 IDSCN NOP
0369 06452 063310R LDA CIDNT GET NEXT IDENT IN SCAN
0370 06453 073306R STA TIDNT SET IDENT ADDRESS FOR IDX
0371 06454 016265R JSB IDX SET IDENT ADDRESS
0372 06455 126451R JMP IDSCN,I RETURN=END OF IDENTs
0373 06456 063311R LDA ID1 GET CURRENT MAIN IDENT ADDRESS
0374 06457 073400R STA IMAIN
0375 06460 063306R LDA TIDNT
0376 06461 073310R STA CIDNT SAVE NEXT IDENT ADDR
0377 06462 163316R LDA ID6,I GET TYPE
0378 06463 001265 RAL,CLE,ERA SET E=M/S
0379 06464 010073 AND M177
0380 06465 053425R CPA PTYPE CURRENT PROGRAM TYPE?
0381 06466 002001 RSS YES
0382 06467 026454R JMP IDSCN+3 NO=GO TRY NEXT IDENT
0383 06470 036451R ISZ IDSCN INCR RETURN ADDRESS
0384 06471 126451R JMP IDSCN,I RETURN
0385*
0386*      GETP DECIPHERS OPERATOR PARAMETER INPUT DATA.
0387*      RESULTS STORED IN PBUF AND A REGISTER INDICATES TERMINATION
0388*      CHARACTER; 0 INDICATES END OF INPUT, COMMA, LEFT PARENTHESIS, AND
0389*      RIGHT PARENTHESIS INDICATES THAT MORE DATA FOLLOWS.
0390*
0391 06472 000000 GETP NOP
0392 06473 063452R LDA BLNKS SET PBUF TO ASCII BLANKS
0393 06474 072214R STA PBUF
0394 06475 072215R STA PBUF+1
0395 06476 072216R STA PBUF+2
0396 06477 062626R LDA NEGPM
0397 06500 002003 SZA,RSS
0398 06501 126472R JMP GETP,I
0399 06502 063563R LDA DFPBF SET CONVERTED CHARACTER STORE
0400 06503 000066 CLE,ELA ADDRESS
0401 06504 073455R STA XPREL
0402 06505 066627R GET10 LDB PMBUF
0403 06506 004065 CLE,ER0 CONVERT TO WORD ADDRESS
0404 06507 160001 LDA B,I GET DATA WORD FROM INPUT BUFFER
0405 06510 002041 SEZ,RSS CHECK IF HI OR LOW CHAR
0406 06511 001727 ALF,ALF SHIFT HI INTO LOW
0407 06512 010074 AND M377 ISOLATE CHARACTER
0408 06513 005600 ELB INCREMENT INPUT BUFFER CHARACTER
0409 06514 006004 INB ADDRESS
0410 06515 076627R STB PMBUF
0411 06516 053536R CPA COMMA COMMA?
0412 06517 026553R JMP GET20 LEFT PARENTHESIS?
0413 06520 053537R CPA LPARN
0414 06521 026553R JMP GET20

```

0415	06522	053540R	CPA RPARN	RIGHT PARENTHESIS?
0416	06523	026553R	JMP GET20	
0417	06524	052556R	CPA SLASH	
0418	06525	026552R	JMP GET15	
0419	06526	067455R	LDB XPREL	GET STORE CHARACTER ADDRESS
0420	06527	004065	CLE,ERB	CONVERT TO WORD ADDRESS
0421	06530	002041	SEZ,RSS	CHECK IF HI OR LOW STORE POSIT.
0422	06531	001727	ALF,ALF	SHIFT INTO HIGH
0423	06532	073371R	STA EXCNT	SAVE CHARACTER
0424	06533	160001	LDA B,I	GET STORED BUFFER WORD
0425	06534	002040	SEZ	
0426	06535	026540R	JMP *+3	
0427	06536	010074	AND M377	SAVE LOW CHARACTER
0428	06537	002001	RSS	
0429	06540	010075	AND M1774	SAVE HI CHARACTER
0430	06541	033371R	IOR EXCNT	IOR IN CURRENT CHARACTER
0431	06542	170001	STA B,I	SAVE IN BUFFER
0432	06543	005600	ELB	
0433	06544	006004	INB	INCREMENT STORE ADDRESS
0434	06545	077455R	STB XPREL	
0435	06546	036626R	ISZ NEGPM	
0436	06547	026505R	JMP GET10	GO TO PROCESS NEXT CHARACTER
0437	06550	002400	CLA	SET A TO INDICATE END
0438	06551	126472R	JMP GETP,I	RETURN
0439	06552	063455R	GET15 LDA SLSHE	
0440	06553	036626R	GET20 ISZ NEGPM	
0441	06554	126472R	JMP GETP,I	
0442	06555	126472R	JMP GETP,I	
0443	06556	000057	SLASH OCT 57	
0444*				
0445*	PROGL CHECKS IF SPECIFIC PROGRAMS WITHIN A FILE ARE TO BE			
0446*	LOADED, THE PROGRAM LIST BEGINS WITH A LEFT PARENTHESIS AND			
0447*	TERMINATES WITH A RIGHT PARENTHESIS, PROGRAMS WITHIN THE FILE			
0448*	ARE SEPARATED BY COMMAS,			
0449*				
0450	06557	000000	PROGL NOP	
0451	06560	002400	CLA	CLEAR PROGRAM NAME COUNTER
0452	06561	073414R	STA PCNTT	
0453	06562	063564R	LDA PLAD	SET PROGRAM NAME BUFFER ADDRESS
0454	06563	072625R	STA PCONT	
0455	06564	063445R	LDA TERMN	CHECK IF AT END OF PARAMETERS
0456	06565	002003	SZA,RSS	
0457	06566	126557R	JMP PROGL,I	END-50 RETURN
0458	06567	053537R	CPA LPARN	CHECK IF LEFT PARENTHESIS
0459	06570	002001	RSS	YES-50 PROGRAMS SPECIFIED
0460	06571	126557R	JMP PROGL,I	NO-50 RETURN
0461	06572	016472R	PROG5 JSB GETP	GET ONE CONTROL PARAMETER
0462	06573	073445R	STA TERMN	
0463	06574	002003	SZA,RSS	
0464	06575	126557R	JMP PROGL,I	
0465	06576	062214R	LDA PBUF	
0466	06577	172625R	STA PCONT,I	SAVE PROG NAME IN BUFFER
0467	06600	036625R	ISZ PCONT	
0468	06601	062215R	LDA PBUF+1	
0469	06602	172625R	STA PCONT,I	
0470	06603	036625R	ISZ PCONT	

PAGE 0087 #06 DOS-III LOADER- UTILITY ROUTINES

0471	06604	062216R	LDA PBUF+2	
0472	06605	010075	AND M1774	
0473	06606	172625R	STA PCONT,I	
0474	06607	036626R	ISZ PCONT	
0475	06610	037414R	ISZ PCNTT	INCREMENT PROGRAM NAME COUNTER
0476	06611	063445R	LDA TERMN	
0477	06612	053540R	CPA RPARN	CHECK IF AT END OF PROG LIST
0478	06613	002001	RSS	YES, SO CHECK IF MORE FOLLOWS
0479	06614	026622R	JMP PROG9	GO CHECK IF AT END OF INPUT
0480	06615	002003	SZA,RSS	
0481	06616	126557R	JMP PROGL,I	RETURN-NO MORE INPUT
0482	06617	016472R	JSB GETP	SKIP OVER NEXT CHARACTER TO GET
0483	06620	073445R	STA TERMN	TRUE TERMINATION SYMBOL
0484	06621	126557R	JMP PROGL,I	RETURN FOR PROGRAM PROCESSING
0485	06622	002002	PROG9 SZA	CHECK IF AT END OF INPUT
0486	06623	026572R	JMP PROG5	NO-GET NEXT PARAMETER
0487	06624	126557R	JMP PROGL,I	YES-SO RETURN
0488	06625	000000	PCONT NOP	
0489	06626	000000	NEGPM NOP	NEG # INPUT CHAR COUNT
0490	06627	000000	PMBUF NOP	INPUT BUFFER CHAR ADDR

0492*

0493* ERROR MESSAGES

0494*

0495 06630 063473R LOVER LDA ERR05

OUTPUT L05 MESSAGE

0496 06631 016344R ABRTX JSB ERROR

0497 06632 063435R ABORT LDA .16

OUTPUT: LOADR TERMINATED MESSAGE

0498 06633 066736R LDB MESAB

0499 06634 016331R JSB SYOUT

0500 06635 026670R JMP LTERM

0501*

0502* SET SIMULATED END-OF-TAPE

0503*

0504 06636 000000 SET NOP

0505 06637 063416R LDA PGMIN

0506 06640 010072 AND M77

0507 06641 002003 SZA,RSS

0508 06642 126636R JMP SET,I

0509 06643 050055 CPA .2

0510 06644 126636R JMP SET,I

0511 06645 033442R IOR M700

0512 06646 073444R STA TEMP

0513 06647 016001X JSB EXEC

SET SIMULATED END OF TAPE

0514 06650 006653R DEF *+3

0515 06651 000056 DEF .3

0516 06652 011444R DEF TEMP

0517 06653 126636R JMP SET,I

```

0519*
0520* THE FOLLOWING ROUTINES PROCESS TWO RECORDS ENTERED
0521* VIA THE BATCH DEVICE, BOTH RECORDS SHOULD CONTAIN
0522* TWO OCTAL NUMBERS (MAX 5 DIGITS EACH) SEPARATED
0523* BY EITHER A BLANK OR A COMMA, IF EITHER OF THESE
0524* NUMBERS IS OMITTED, THE DEFAULT SYSTEM VALUE WILL
0525* BE ASSUMED. THE FIRST RECORD IS INTERPRETED AS
0526* LOWER & UPPER BASE PAGE BOUNDS. THE SECOND
0527* RECORD IS INTERPRETED AS LOWER & UPPER MAIN
0528* PROGRAM BOUNDS.
0529*
0530* IF THE BATCH DEVICE IS THE SYSTEM CONSOLE, INPUT
0531* OF THE RECORDS WILL BE PROMPTED BY:
0532*
0533*          BP   BND (L,U)?
0534*          PROG BND (L,U)?
0535*
0536* AN ERROR IN THE FIRST RECORD WILL RESULT IN AN
0537* L18 ERROR; AN ERROR IN THE SECOND AN L19. FOL-
0538* LOWING THE ERROR, THE RECORD MAY BE RE-ENTERED.
0539*
0540* IF THE BATCH DEVICE IS NOT THE SYSTEM CONSOLE,
0541* INPUT OF THE RECORDS SHOULD IMMEDIATELY FOLLOW
0542* THE :PR,LOADR DIRECTIVE, ANY ERROR IN EITHER WILL
0543* CAUSE THE LOADER TO ABORT.
0544*
0545 06654 062715R STBND LDA BMES      SET FOR
0546 06655 006400      CLB           BASE PAGE BOUNDS
0547 06656 016737R     JSB STB       ENTER & PROCESS BOUNDS
0548 06657 026675R     JMP BPBE      ERROR RETURN
0549 06660 073456R     STA UBFWA     "A" = LOWER BP BOUND
0550 06661 077457R     STB UBLWA     "B" = UPPER BP BOUND
0551 06662 007004      CMB,INB      MAKE SURE BOUNDS ARE
0552 06663 046714R     ADB B1777    ON THE BASE PAGE
0553 06664 006020      SSB          ( < 2000B)
0554 06665 026675R     JMP BPBE     NO = ERROR
0555*
0556 06666 062726R STB1 LDA BMES1    SET FOR
0557 06667 006404      CLB,INB      MAIN PROGRAM BOUNDS
0558 06670 016737R     JSB STB       ENTER & PROCESS BOUNDS
0559 06671 026703R     JMP PBE      ERROR RETURN
0560 06672 073460R     STA UMFWA     "A" = LOWER MAIN PROG BOUND
0561 06673 077461R     STB UMLWA     "B" = UPPER MAIN PROG BOUND
0562 06674 026034R     JMP LOAD1    RETURN TO MAIN PROCESSING

0564* ERROR ROUTINES

0566 06675 062712R BPBE LDA ERR18    BASE PAGE
0567 06676 016344R     JSB ERROR     BOUND ERROR = L18
0568 06677 060111      LDA BATCH     INPUT FROM
0569 06700 050112      CPA SYSTY     SYSTEM CONSOLE?
0570 06701 026654R     JMP STBND     YES = REENTER
0571 06702 026632R     JMP ABORT     NO = ABORT

0573 06703 062713R PBE  LDA ERR19    MAIN PROGRAM

```

PAGE 0090 #06 DOS-III LOADER- UTILITY ROUTINES

0574	06704	016344R	JSB ERROR	BOUND ERROR - L19
0575	06705	060111	LDA BATCH	INPUT FROM
0576	06706	050112	CPA SYSTY	SYSTEM CONSOLE?
0577	06707	026666R	JMP STB1	YES = REENTER
0578	06710	026632R	JMP ABORT	NO = ABORT

0580* CONSTANTS,ETC, FOR PROGRAM BOUNDS ROUTINES

0582	06711	177720	NB60	OCT -60	
0583	06712	030470	ERR18	ASC 1,18	
0584	06713	030471	ERR19	ASC 1,19	
0585	06714	001777	B1777	OCT 1777	
0586	06715	006716R	BMES	DEF ++1	
0587	06716	041120		ASC 8,BP	BND (L,U)?
0588	06726	006727R	BMES1	DEF ++1	
0589	06727	050122		ASC 8,PROG	BND (L,U)?

PAGE 0091 #06 DOS-III LOADER- UTILITY ROUTINES

0591* ROUTINE TO ENTER TWO ASCII NUMBERS FROM THE BATCH
0592* DEVICE AND CONVERT THEM TO OCTAL.

0593*

0594* UPON ENTRY:

0595*

0596* "A" = BUFFER ADDRESS OF MESSAGE TO BE
0597* OUTPUT IF BATCH IS SYSTEM CONSOLE

0598* "B" = BASE PAGE/MAIN PROGRAM FLAG

0599* 0 => BASE PAGE

0600* 1 => MAIN PROGRAM

0601*

0602* ON EXIT:

0603*

0604* P+1 : ERROR RETURN
0605* ILLEGAL OCTAL DIGIT
0606* LOWER > UPPER BOUND
0607* NUMBER > 5 DIGITS

0608*

0609* P+2 : NORMAL RETURN
0610* "A" = LOWER BOUND
0611* "B" = UPPER BOUND

0612*

0613* PARAMETERS USED:

0614*

0615* DATAB = INPUT BUFFER
0616* PLIST = 3-WORD ASCII BUFFER FOR CONV
0617* ID1 = POINTER INTO PLIST
0618* ID2 = SUM FOR OCTAL NUMBER CONVERSION
0619* ID3 = BASE PAGE/MAIN PROG FLAG
0620* ID4 = LOWER BOUND
0621* ID5 = LOWER/UPPER BOUND FLAG
0622* ID6 = POINTER INTO DATAB
0623* ID7 = UPPER/LOWER FLAG FOR DATAB
0624* LST1 = UPPER/LOWER FLAG FOR PLIST
0625* LST2 = IGNORE BLANKS FLAG

0626*

0627 06737 000000 STB NOP
0628 06740 072756R STA BBADD STORE MESSAGE BUFFER ADDRESS
0629 06741 077313R STB ID3 SAVE BP/MAIN PROG FLAG

0630*

0631 06742 060111 STB2 LDA BATCH IF BATCH IS
0632 06743 050112 CPA SYSTY SYSTEM CONSOLE, SET
0633 06744 033544R IOR M400 ECHO BIT IN
0634 06745 073444R STA TEMP CONTROL WORD
0635 06746 060111 LDA BATCH IF BATCH IS
0636 06747 050112 CPA SYSTY NOT SYSTEM CONSOLE,
0637 06750 002001 RSC SKIP OUTPUTTING OF
0638 06751 026760R JMP STB3 MESSAGE

0639*

0640 06752 016001X JSB EXEC OUTPUT MESSAGE TO SYSTEM CONSOLE
0641 06753 006760R DEF ++5
0642 06754 000055 DEF .2
0643 06755 000054 DEF .1
0644 06756 000000 BBADD NOP BUFFER ADDRESS STORED HERE
0645 06757 000063 DEF .8

0646*

PAGE 0092 #06 DOS-III LOADER- UTILITY ROUTINES

0647	06760	016001X	STB3	JSB EXEC	INPUT TWO ASCII NUMBERS
0648	06761	006766R		DEF ++5	INTO DATAB
0649	06762	000054		DEF ,1	
0650	06763	011444R		DEF TEMP	
0651	06764	011207R		DEF DATAB	
0652	06765	007427R		DEF N72	
0653	06766	006003		SZB,RSS	
0654	06767	026742R		JMP STB2	TRY AGAIN IF NO CHAR INPUT
0655	06770	007004		CMB,INB	
0656	06771	076626R		STB NEGPM	SAVE -# CHARS INPUT
0657*					
0658	06772	002400		CLA	SET FOR
0659	06773	073315R		STA ID5	LOWER BOUND
0660	06774	062273R		LDA DATB	START SCAN AT BEGINNING
0661	06775	073316R		STA ID6	OF DATAB
0662	06776	003400		CCA	SET UPPER/LOWER FLAG
0663	06777	073317R		STA ID7	TO UPPER
0664*					
0665	07000	002400	STB4	CLA	CLEAR
0666	07001	072000R		STA PLIST	ASCII BUFFER
0667	07002	072001R		STA PLIST+1	
0668	07003	072002R		STA PLIST+2	
0669	07004	063564R		LDA PLAD	INITIALIZE ASCII
0670	07005	073311R		STA ID1	BUFFER POINTER
0671	07006	003400		CCA	SET UPPER/LOWER FLAG
0672	07007	073300R		STA LST1	TO UPPER
0673	07010	073301R		STA LST2	SET TO IGNORE BLANKS
0674*					
0675	07011	067317R	STB5	LDB ID7	GET CHARACTER
0676	07012	163316R		LDA ID6,I	
0677	07013	006002		SZB	
0678	07014	001727		ALF,ALF	
0679	07015	010073		AND M177	
0680	07016	007002		CMB,SZB	
0681	07017	037316R		ISZ ID6	
0682	07020	077317R		STB ID7	SET NEW UPPER/LOWER FLAG
0683*					
0684	07021	053453R		CPA BLANK	IS IT A BLANK?
0685	07022	027061R		JMP PRBLN	YES - PROCESS BLANK
0686	07023	053536R		CPA COMMA	IS IT A COMMA?
0687	07024	027055R		JMP PRCPA	YES - PROCESS COMMA
0688*					
0689	07025	066711R		LDB NB60	MAKE SURE VALID OCTAL DIGIT
0690	07026	044000		ADB A	
0691	07027	006020		SSB	
0692	07030	126737R		JMP STB,I	<60 ERROR
0693	07031	044043		ADB N8	
0694	07032	006021		SSB,RSS	
0695	07033	126737R		JMP STB,I	>67 ERROR
0696	07034	073301R		STA LST2	<ST2>0, SO DON'T IGNORE BLANKS
0697*					
0698	07035	067300R		LDB LST1	STORE CHARACTER IN ASCII BUFFER
0699	07036	006002		SZB	
0700	07037	001727		ALF,ALF	UPPER CHARACTER
0701	07040	133311R		IOR ID1,I	
0702	07041	173311R		STA ID1,I	


```

0703 07042 006002      SZB
0704 07043 027050R     JMP SCHK
0705 07044 063564R     LDA PLAD
0706 07045 040055      ADA ,2
0707 07046 053311R     CPA ID1
0708 07047 126737R     JMP STB,I      6 DIGITS - ERROR
0709 07050 007002 SCHK CMB,SZB
0710 07051 037311R     ISZ ID1
0711 07052 077300R     STB LST1
0712*
0713 07053 017070R     JSB DONE      DONE?
0714 07054 027011R     JMP STB5      NO- GET NEXT CHARACTER
0715*
0716 07055 003400 PRMA CCA      SET TO
0717 07056 073301R     STA LST2      IGNORE BLANKS
0718 07057 017070R     JSB DONE      DONE?
0719 07060 027066R     JMP PRN      NO- CONVERT # & CONTINUE
0720*
0721 07061 063301R PRBLN LDA LST2  IGNORE
0722 07062 002021      SSA,RSS      BLANKS ?
0723 07063 027066R     JMP PRN      NO - CONVERT #
0724 07064 017070R     JSB DONE      YES- DONE?
0725 07065 027011R     JMP STB5      NO- GET NEXT CHAR
0726*
0727 07066 017101R PRN JSB CONV      CONVERT ASCII TO OCTAL
0728 07067 027000R     JMP STB4      RETURN IF 2ND NUMBER NEEDED

```

```

0730* ROUTINE TO TEST WHETHER OR NOT ALL CHARACTERS
0731* INPUT HAVE BEEN PROCESSED
0732*
0733* RETURN:      P+1 - MORE CHARACTERS TO PROCESS
0734* NO RETURN IF ALL CHARACTERS HAVE BEEN PROCESSED.
0735*      NUMBER IN ASCII BUFFER WILL BE CONVERTED.
0736*      IF ONLY ONE NUMBER HAS BEEN ENTERED,
0001*      2ND NUMBER WILL BE DEFAULT VALUE.
0002*

```

```

0003 07070 000000 DONE NOP
0004 07071 036626R     ISZ NEGPM      MORE CHARACTERS?
0005 07072 127070R     JMP DONE,I      YES- CONTINUE
0006 07073 017101R DONE1 JSB CONV      CONVERT # IN ASCII BUFFER
0007 07074 002400      CLA      RETURN IF 2ND NUMBER NEEDED
0008 07075 072000R     STA PLIST      CLEAR BUFFER FOR
0009 07076 072001R     STA PLIST+1    DEFAULT VALUE
0010 07077 072002R     STA PLIST+2
0011 07100 027073R     JMP DONE1      CONVERT 2ND NUMBER

```

```

0013* ROUTINE TO CONVERT NUMBER IN ASCII BUFFER TO OCTAL.
0014* INPUT IS TERMINATED BY A NULL. IF VALUE=0,
0015* THE SYSTEM DEFAULT VALUE IS ASSUMED.
0016* THE FIRST TIME THIS ROUTINE IS CALLED, IT CONVERTS
0017* THE LOWER BOUND, SAVES IT AND RETURNS. THE SECOND
0018* TIME, IT CONVERTS THE UPPER BOUND, THEN SETS

```

```

0019*
0020*      "A" = LOWER BOUND
0021*      "B" = UPPER BOUND
0022* AND RETURNS TO P+2 OF STB ROUTINE.
0023*
0024 07101 000000 CONV NOP
0025 07102 063564R LDA PLAD      INITIALIZE ASCII BUFFER
0026 07103 073311R STA ID1      POINTER
0027 07104 006400 CLB          INITIALIZE OCTAL
0028 07105 077312R STB ID2      SUM TO ZERO
0029*
0030 07106 163311R CONV1 LDA ID1,I GET 1ST CHAR
0031 07107 001727 ALF,ALF
0032 07110 010073 AND M177
0033 07111 017117R JSB CON      CONVERT IT
0034 07112 163311R LDA ID1,I GET 2ND CHAR
0035 07113 010073 AND M177
0036 07114 017117R JSB CON      CONVERT IT
0037 07115 037311R ISZ ID1      SET FOR NEXT WORD
0038 07116 027106R JMP CONV1

0040 07117 000000 CON NOP
0041 07120 002003 SZA,RSS      IS CHAR NULL?
0042 07121 027131R JMP CONV2    YES = INPUT TERMINATED
0043 07122 010062 AND ,7      N= MASK OFF 600
0044*
0045 07123 067312R LDB ID2      UPDATE OCTAL SUM
0046 07124 005222 RBL,RBL
0047 07125 005200 RBL
0048 07126 044000 ADB A
0049 07127 077312R STB ID2
0050 07130 127117R JMP CON,I
0051*
0052 07131 006002 CONV2 SZB
0053 07132 027161R JMP CONV3    DEFAULT BOUNDS NECESSARY?
0054*                                NO= RET GIVEN BOUNDS
0055 07133 063315R LDA ID5      PROCESSING
0056 07134 002002 SZA          LOWER BOUND?
0057 07135 027145R JMP UPPR      NO = UPPER
0058 07136 064255 LDB SBFWA    SET FOR BASE PAGE LOWER
0059 07137 063313R LDA ID3      IS IT FOR
0060 07140 002002 SZA          BASE PAGE?
0061 07141 064254 LDB SMFWA    NO = CHANGE TO MAIN PRG LOWER
0062 07142 077314R LWR STB ID4    SAVE LOWER BOUND
0063 07143 077315R STB ID5      SET FOR UPPER BOUND NEXT
0064 07144 127101R JMP CONV,I    RETURN
0065*
0066 07145 064256 UPPR LDB SBLWA  SET FOR BASE PAGE UPPER
0067 07146 063313R LDA ID3      IS IT FOR BASEE PAGE?
0068 07147 002002 SZA
0069 07150 064100 LDB SMLWA    NO = CHANGE TO MAIN PROG UPPE
0070*
0071 07151 063314R UPPR1 LDA ID4  MAKE SURE LOWER <= UPPER
0072 07152 003004 CMA,INA
0073 07153 040001 ADA B

```

PAGE 0095 #07 DOS=III LOADER- UTILITY ROUTINES

0074	07154	002020	SSA	
0075	07155	126737R	JMP STB,I	NO - ERROR RETURN
0076*				
0077	07156	063314R	LDA ID4	"A" = LOWER BOUND
0078	07157	036737R	ISZ STB	"B" = UPPER BOUND
0079	07160	126737R	JMP STB,I	RETURN TO P+2 OF STB
0080*				
0081	07161	063315R CONV3	LDA ID5	IS THIS FOR LOWER BOUND?
0082	07162	002002	SZA	
0083	07163	027151R	JMP UPPR1	NO
0084	07164	027142R	JMP LWR	YES

0086*
 0087* THE FOLLOWING ROUTINES LINK A PROGRAM THROUGH
 0088* CURRENT PAGE LINKS.
 0089*
 0090* ALINK = CALLED AFTER NAM RECORD IS READ.
 0091* DETERMINES WHETHER OR NOT CURRENT
 0092* PAGE LINKS ARE TO BE USED. SETS UP
 0093* LOW LINK AREA (BOTH INITIAL GUESS &
 0094* FIXED LENGTH FOLLOWING PASS 1)
 0095*
 0096* BLINK = CALLED AFTER END RECORD IS READ.
 0097* IF UPPER LINK AREA IS PRESENT, IT IS
 0098* OUTPUT TO THE DISC.
 0099*
 0100* DLINK = CALLED FROM BPSCN AFTER FAILING TO
 0101* FIND DESIRED LINK IN BASE PAGE, BUT
 0102* PRIOR TO ALLOCATING A NEW BASE PAGE
 0103* LINK. IF NO CURRENT PAGE LINK IS POS-
 0104* SIBLE, RETURN IS MADE TO P+1 & A BP
 0105* LINK IS ALLOCATED. IF A CURRENT PAGE
 0106* LINK IS POSSIBLE, IT IS CREATED &
 0107* RETURN IS MADE TO LINKC+1.
 0108*
 0109* CROSS = CALLED FROM DLINK WHEN A PAGE
 0110* BOUNDARY IS CROSSED OR FROM END
 0111* RECORD DETECTED BEFORE PASS 3. FIRST
 0112* PAGE CROSSING SETS FOR PASS 2 & FIXES
 0113* LENGTH OF LOW LINK AREA. SECOND PAGE
 0114* CROSSING SETS FOR PASS 3 & OUTPUTS
 0115* LOW LINK AREA TO DISC. THIRD PAGE
 0116* CROSSING RESULTS IN DEFAULTING LINK
 0117* TO BASE PAGE.
 0118*

0120	07166	000000	ALINK	NOP	
0121	07166	063420R	LDA	L01	CURRENT PAGE
0122	07167	002020	SSA		LINKS POSSIBLE?
0123	07170	127165R	JMP	ALINK,I	NO - RETURN
0124	07171	002002	SZA		YES = PASS 3?
0125	07172	027244R	JMP	PS3	YES = SET LOW LINK POINTER
0127	07173	063417R	LDA	L001	PASS 1 = FOR LOW LINK
0128	07174	002002	SZA		AREA GUESS?
0129	07175	027234R	JMP	PS2	NO = PASS 2 FOR FIXED LOW LNK
0131	07176	063416R	LDA	CPL	CURRENT PAGE LINKS REQUESTED
0132	07177	002003	SZA,RSS		IN CALLING SEQUENCE?
0133	07200	027232R	JMP	NCPL	NO = SET L01 < 0 FOR NO CPL
0135	07201	063603R	LDA	TPREL	GET ADDRESS
0136	07202	070001	STA	B	OF LAST WORD
0137	07203	033444R	IOR	01777	OF PAGE
0138*					
0139	07204	007004	CMB,INB		COMPUTE #
0140	07205	006004	INB		WORDS

PAGE 0097 #07 DOS-III LOADER- UTILITY ROUTINES

0141	07206	044000		ADB A	REMAINING
0142	07207	077411R		STB TEMP2	ON PAGE
0143*					
0144	07210	063421R		LDA PLGTH	COMPUTE # WORDS
0145	07211	001265		RAL,CLE,ERA	OF PROGRAM
0146	07212	007004		CMB,INB	THAT FALL
0147	07213	044000		ADB A	BEYOND THIS
0148	07214	077410R		STB TEMP1	PAGE
0149*					
0150	07215	006020		SSB	IF PROGRAM FITS ON ONE PAGE
0151	07216	027232R		JMP NCPL	CURRENT PAGE LINKS ARE NOT
0152	07217	006003		SZB,RSS	REQUIRED.
0153	07220	027232R		JMP NCPL	
0154*					
0155	07221	063411R		LDA TEMP2	COMPUTE MINIMUM OF:
0156	07222	001100		ARS	HALF # WORDS OF PROGRAM
0157	07223	007004		CMB,INB	ON CURRENT PAGE -OR-
0158	07224	044000		ADB A	# WORDS OF PROGRAM
0159	07225	006021		SSB,RSS	ON NEXT PAGE
0160*					
0161	07226	063410R		LDA TEMP1	DIVIDE THIS
0162	07227	001121		ARS,ARS	MINIMUM BY FOUR
0163	07230	002002		SZA	IF NON-ZERO,USE AS SIZE OF LOW
0164	07231	027235R		JMP PS1	LINK AREA, IF EQUAL TO ZERO,
0166	07232	003400	NCPL	CCA	SET L01 < 0 TO INDICATE
0167	07233	027026R		JMP LOADX	NO CURRENT PAGE LINKS
0169	07234	063413R	PS2	LDA LWH2	PASS 2 - FIXED LENGTH OF LOW LNK
0170	07235	067603R	PS1	LDB TPREL	SET NEW:
0171	07236	077410R		STB TEMP1	LOW LIMIT OF LOW LINK AREA
0172	07237	044000		ADB A	(ADD LENGTH OF LINK AREA)
0173	07240	077603R		STB TPREL	LOAD ADDRESS
0174	07241	077424R		STB CPMAX	UPPER LIMIT OF LOW LINK AREA
0175	07242	003004		CMA,INA	SAVE NEGATIVE LENGTH OF LINK
0176	07243	073413R		STA LWH2	AREA FOR OUTPUT TO DISC (BLIND
0178	07244	063410R	PS3	LDA TEMP1	SET UP POINTERS
0179	07245	073423R		STA CPLOC	FOR LOW LINK AREA
0180	07246	073422R		STA CPBAS	
0181	07247	013446R		AND M0760	
0182	07250	073425R		STA CPADD	GET CURRENT PAGE ADDRESS
0183	07251	127165R		JMP ALINK,I	

0185	07252	000000	DLINK	NOP	
0186	07253	062601R		LDA BPSCN	WAS CALL FROM
0187	07254	053421R		CPA CLINK	DBLBPROCESSOR?
0188	07255	002001		RSS	
0189	07256	127252R		JMP DLINK,I	NO - USE BASE PAGE LINK
0190	07257	063420R		LDA L01	USING CURRENT
0191	07260	002020		SSA	PAGE LINKS?
0192	07261	127252R		JMP DLINK,I	NO - USE BASE PAGE LINK
0194	07262	063425R	SETL	LDA CPADD	GET LINK PAGE
0195	07263	053412R		CPA PAGNO	SAME AS INSTRUCTION PAGE?
0196	07264	002001		RSS	
0197	07265	027317R		JMP CROSS	NO - PAGE BOUNDARY CROSSING
0199	07266	063277R		LDA PLST	SET TO
0200	07267	073356R		STA BLINK	BOTTOM OF LINK AREA
0201	07270	067422R		LDB CPBAS	GET NUMBER OF CURRENT PAGE LINKS
0202	07271	007004		CMB,INB	ALLOCATED SO FAR. IF ZERO,
0203	07272	047423R		ADB CPLOC	JUMP TO ALLOCATE.
0204	07273	007007		CMB,INB, SZB, RSS	
0205	07274	027303R		JMP DLIN1	
0206	07275	163356R	DLIN2	LDA BLINK,I	IF NOT ZERO, SCAN CURRENT PAGE
0207	07276	053411R		CPA OPRND	LINKS PRESENT FOR POSS. MATCH
0208	07277	027303R		JMP DLIN1	MATCH FOUND- RETURN ADDRESS
0209	07300	037356R		ISZ BLINK	MATCH NOT FOUND- CONTINUE
0210	07301	006006		INB, SZB	SEARCHING THROUGH LINKS
0211	07302	027275R		JMP DLIN2	
0213	07303	063356R	DLIN1	LDA BLINK	MAKE SURE CORE BUFFER DOESN'T
0214	07304	053307R		CPA PIDNT	OVERFLOW INTO ID TABLE
0215	07305	127252R		JMP DLINK,I	YES : USE BP LINKS
0216	07306	063423R		LDA CPLOC	GET NEXT AVAIL LINK LOC
0217	07307	040001		ADA B	ADD OFFSET IN CASE NOT NEW LINK
0218	07310	053424R		CPA CPMAX	OVERFLOW LINK AREA?
0219	07311	127252R		JMP DLINK,I	YES- MUST USE BP LINK
0220	07312	006003		SZB, RSS	IF NEW LINK, INCREMENT THE
0221	07313	037423R		ISZ CPLOC	LOCATION COUNTER
0222	07314	067411R		LDB OPRND	GET THE LINK WORD
0223	07315	177356R		STB BLINK,I	SET IN THE LINK AREA
0224	07316	026106R		JMP LINKC+1	RETURN TO DBL PROCESSING
0226	07317	063420R	CROSS	LDA L01	PASS 3?
0227	07320	002002		SZA	
0228	07321	027332R		JMP BLIN3	YES=ASSIGN UPPR LINK OR RETURN
0230	07322	063417R		LDA L001	PASS 1?
0231	07323	002003		SZA, RSS	
0232	07324	027350R		JMP FLLA	YES=FIX LOW LINK AREA
0234	07325	063413R		LDA LWH2	IF LENGTH OF LOW LINK AREA IS
0235	07326	002002		SZA	GREATER THAN ZERO,
0236	07327	017373R		JSB BLIN0	OUTPUT IT TO DISC
0237	07330	002404	BLIN2	CLA, INA	SET FOR PASS 3
0238	07331	027026R		JMP LOADX	AND GO

PAGE 0099 #07 DOS-III LOADER- UTILITY ROUTINES

0240	07332	002001	BLIN3	RSS	SKIP THE 1ST TIME THROUGH, THEN
0241	07333	127252R		JMP DLINK,I	RSS CHANGED TO NOP - (BP LINKS)
0243	07334	063421R		LDA PLGTH	COMPUTE
0244	07335	001265		RAL,CLE,ERA	
0245	07336	043603R		ADA TPREL	
0246	07337	073423R		STA CPLOC	
0247	07340	073422R		STA CPBAS	LOW BOUND OF UPPER LINK AREA
0248	07341	013446R		AND M0760	
0249	07342	073425R		STA CPADD	CURRENT PAGE ADDRESS
0250	07343	043445R		ADA M2000	
0251	07344	073424R		STA CPMAX	UPPER BOUND OF UPPER LINK AREA
0252	07345	002400		CLA	CLEAR BLIN3, SO SUBSEQUENT PAGE
0253	07346	073332R		STA BLIN3	CROSSINGS WILL USE BP LINKS
0254	07347	027262R		JMP SETL	RETURN TO ALLOCATE CP LINK
0256	07350	063410R	FLLA	LDA TEMP1	COMPUTE # LINKS ACTUALLY USED IN
0257	07351	003004		CMA,INA	LOW LINK AREA, & SAVE FOR
0258	07352	043423R		ADA CPLOC	FIXED LENGTH.
0259	07353	073413R		STA LWH2	
0260	07354	002404		CLA,INA	SET FOR PASS 2
0261	07355	027011R		JMP LD3	AND GO
0263	07356	000000	BLINK	NOP	
0264	07357	037412R		ISZ LWH1	IF FROM END RECORD BEFORE PASS 3
0265	07360	127356R		JMP BLINK,I	NO UPPER LINK AREA, RETURN
0267	07361	063423R		LDA CPLOC	COMPUTE NEGATIVE #
0268	07362	003004		CMA,INA	OF WORDS IN UPPER LINK AREA.
0269	07363	043603R		ADA TPREL	(TPREL IS AT END OF CURRENT
0270	07364	073413R		STA LWH2	PROGRAM)
0271	07365	002003		SZA,RSS	IF LENGTH = 0, DON'T
0272	07366	127356R		JMP BLINK,I	OUTPUT
0273	07367	017373R		JSB BLIN0	OTHERWISE) OUTPUT UPPER LINK2
0275	07370	063423R		LDA CPLOC	ADJUST NEW RELOCATION ADDRESS TO
0276	07371	073603R		STA TPREL	BEYOND UPPER LINK AREA
0277	07372	127356R		JMP BLINK,I	
0279	07373	000000	BLIN0	NOP	
0280	07374	063277R		LDA PLST	SET POINTER TO START
0281	07375	073165R		STA ALINK	OF LINK BUFFER
0282	07376	067422R		LDB CPBAS	SET POINTER TO START OF
0283	07377	077426R		STB MAXC	CORE LOCATION
0284*					
0285	07400	163165R	BLIN1	LDA ALINK,I	GET THE LINK WORD
0286	07401	067426R		LDB MAXC	OUTPUT LINK TO
0287	07402	017706R		JSB LABD0	DISC BUFFER
0288	07403	037426R		ISZ MAXC	INCREMENT CORE ADDR POINTER
0289	07404	037165R		ISZ ALINK	INCREMENT LINK BUFFER POINTER
0290	07405	037413R		ISZ LWH2	LWH2 * = # WORDS IN BUFFER
0291	07406	027400R		JMP BLIN1	MORE TO OUTPUT
0292	07407	127373R		JMP BLIN0,I	DONE

PAGE 0100 #07 DOS-III LOADER- UTILITY ROUTINES

0294	07410	000000	TEMP1	NOP	LOW BOUND OF LOW LINK AREA
0295	07411	000000	TEMP2	NOP	TEMPORARY
0296	07412	000000	LWH1	NOP	UPPER LINK AREA PRESENT FLAG
0297	07413	000000	LWH2	NOP	LENGTH OF CURRENT PAGE LINK AREA
0298	07414	000000	LWH3	NOP	HOLDS ORIGINAL TBREL
0299	07415	000000	LWH4	NOP	HOLDS ORIGINAL TPREL
0300	07416	000000	CPL	NOP	CURRENT PAGE LINK FLAG =0,CP LNK
0301	07417	000000	L001	NOP	PASS 1/2 FLAG
0302	07420	000000	L01	NOP	PASS 3 FLAG
0303	07421	004105R	CLINK	DEF LINKC	
0304	07422	000000	CPBAS	NOP	LOW BOUND OF CURRENT PAGE BUFFER
0305	07423	000000	CPLOC	NOP	CURRENT POINTER IN CP BUFFER
0306	07424	000000	CPMAX	NOP	UPPER BND OF CURRENT PAGE BUFFER
0307	07425	000000	CPADD	NOP	CURRENT PAGE ADDRESS OF BUFFER
0308	07426	000000	MAXC	NOP	MAX RELOCATION ADDRESS

PAGE 0101 #07 DOS=III LOADER - CONSTANTS & VARIABLES

0310	07427	177670	N72	DEC	-72	
0311	07430	177755	N19	DEC	-19	
0312	07431	177756	N18	DEC	-18	
0313	07432	000013	.11	DEC	11	
0314	07433	000014	.12	DEC	12	
0315	07434	000016	.14	DEC	14	
0316	07435	000020	.16	DEC	16	
0317	07436	000022	.18	DEC	18	
0318	07437	000044	.36	DEC	36	
0319	07440	000177	.127	DEC	127	
0320	07441	000600	M600	OCT	600	
0321	07442	000700	M700	OCT	700	
0322	07443	001100	M1100	OCT	1100	
0323	07444	001777	O1777	OCT	1777	
0324	07445	002000	M2000	OCT	2000	
0325	07446	076000	M0760	OCT	076000	
0326	07447	174000	M1740	OCT	174000	
0327	07450	100000	MSIGN	OCT	100000	
0328	07451	020052	BLAST	OCT	20052	
0329	07452	020040	BLNKS	OCT	20040	
0330	07453	000040	BLANK	OCT	40	
0331	07454	020000	U0LNK	OCT	20000	
0332*						
0333	07455	027505	SLSHE	ASC	1,/E	CONTROL PARAMETER TERMINATOR
0334	07456	007457R	DADRR	DEF	++1	
0335	07457	022104	SDBP1	ASC	2,SDBP	SDBP1
0336	07461	030400		OCT	030400	
0337	07462	042105	CHRDE	ASC	2,DEBU	DEBUG
0338	07464	043400		OCT	043400	
0339	07465	022104	SDBP2	ASC	2,SDBP	SDBP2
0340	07467	031000		OCT	031000	
0341	07470	022115	\$MEMR	ASC	2,\$MEM	\$MEMR
0342	07472	051000		OCT	051000	
0343	07473	042102	CHRDB	ASC	2,DBKP	DBKPT
0344	07475	052000		OCT	052000	
0345	07476	042530	AEXEC	ASC	2,EXEC	
0346	07500	020000		OCT	020000	
0347	07501	007502R	D0SHD	DEF	++1	
0348	07502	051105		ASC	9,RELOCATING LOADER	
0349	07513	007514R	NAENT	DEF	++1	
0350	07514	047101		ASC	10,NAME	PROG BOUNDS BP BOUNDS
0351*						
0352	07536	000054	COMMA	OCT	54	COMMA
0353	07537	000050	LPARN	OCT	50	LEFT PARENTHESIS
0354	07540	000051	RPARN	OCT	51	RIGHT PARENTHESIS
0355	07541	000060	M60	OCT	60	
0356	07542	000200	M200	OCT	200	
0357	07543	000300	M300	OCT	300	
0358	07544	000400	M400	OCT	400	
0359	07545	177600	M7600	OCT	177600	
0360	07546	000071	D57	DEC	57	
0361	07547	177600	N128	DEC	-128	
0362	07550	000200	.128	DEC	128	
0363*						
0364	07551	007607R	ADBUF	DEF	DBUF	ADDRESS OF DBUF BUFFER
0365	07552	010007R	AIBUF	DEF	IBUF	

0366	07553	010207R	ALBUF	DEF	LBUF	
0367	07554	011260R	AMEM5	DEF	MLIST+5	
0368	07555	011263R	AMEM8	DEF	MLIST+8	
0369	07556	011267R	AME12	DEF	MLIST+12	
0370	07557	011272R	AME15	DEF	MLIST+15	
0371	07560	011253R	AMLST	DEF	MLIST	
0372	07561	000157	DFUTS	DEF	UDNTS	
0373	07562	000160	DFSTS	DEF	SYNTS	
0374	07563	000214R	DFPBF	DEF	PBUF	ADDRESS OF NAME BUFFER
0375	07564	000000R	PLAD	DEF	PLIST	
0376*						
0377	07565	007566R	DPWRS	DEF	++1	
0378	07566	010000		OCT	10000	
0379	07567	001000		OCT	1000	
0380	07570	000100		OCT	100	
0381	07571	000010		OCT	10	
0382	07572	000001		OCT	1	
0383*						
0384	07573	007574R	DPWRS	DEF	++1	
0385	07574	023420		DEC	10000	
0386	07575	001750		DEC	1000	
0387	07576	000144		DEC	100	
0388	07577	000012		DEC	10	
0389	07600	000001		DEC	1	
0390*						
0391	07601	007602R	RBAD	DEF	++1	RELOCATION BASE TABLE
0392	07602	000000		OCT	0	ABSOLUTE PROGRAM BASE
0393	07603	000000	TPREL	OCT	0	CURRENT PROGRAM BASE
0394	07604	000000		OCT	0	BP RELOCATION ADDRESS
0395	07605	000000	COMAD	OCT	0	CURRENT COMMON RELOCATION BASE
0396	07606	000000		OCT	0	ABSOLUTE
0397*						
0398	07607	000000	DBUF	BSS	128	RELOCATABLE DISC INPUT BUFFER
0399	10007	000000	IBUF	BSS	128	ID SEGMENT BUFFER
0400	10207	000000	LBUF	BSS	512	
0401	00000		PLIST	EQU	LOADR	PUT 128 WD BUFFER AT START OF LO
0402	11207	000000	DATAB	BSS	36	PARAMETER INPUT BUFFER
0403	11253	000000	MLIST	BSS	18	MEMORY MAP BUFFER
0404	00210		TBUF	EQU	LOADR+136	TEMP BUFFER
0405	00214		PBUF	EQU	LOADR+140	
0406*						
0407	11275	000000	BLST	NOP		LOADER SYMBOL TABLE ADDRESS
0408	11276	000000	TLST	NOP		CURRENT LST ADDRESS
0409	11277	000000	PLST	NOP		END LST ADDRESS
0410*						
0411	11300	000000	LST1	NOP		LST NAME 1,2
0412	11301	000000	LST2	NOP		LST NAME 3,4
0413	11302	000000	LST3	NOP		LST NAME 5 ; ORDINAL
0414	11303	000000	LST4	NOP		PROGRAM IDENT ADDRESS
0415	11304	000000	LST5	NOP		BASE PAGE LINK ADDRESS
0416*						
0417	11305	000000	BIDNT	NOP		PROGRAM NAME TABLE ADDRESS
0418	11306	000000	TIDNT	NOP		CURRENT IDENT ADDRESS
0419	11307	000000	PIDNT	NOP		END IDENT ADDRESS
0420	11310	000000	CIDNT	NOP		WORKING IDENT ADDRESS
0421*						

PAGE 0103 #07 DOS-III LOADER - CONSTANTS & VARIABLES

0422	11311	000000	ID1	NOP	IDENT NAME 1,2
0423	11312	000000	ID2	NOP	IDENT NAME 3,4
0424	11313	000000	ID3	NOP	IDENT NAME 5 ; USEAGE FLAG
0425	11314	000000	ID4	NOP	IDENT COMMON LENGTH
0426	11315	000000	ID5	NOP	IDENT DISC ADDRESS
0427	11316	000000	ID6	NOP	(15)=MAIN/SUBROUTINE; (6=0)=TYPE
0428	11317	000000	ID7	NOP	DISC LENGTH; MAIN IDENT FOR SEGM
0429*					
0430	11320	000000	ABSID	NOP	IDENT ADDR FOR NEXT SEGMENT SCAN
0431	11321	000000	AMAD	NOP	CURRENT MLIST ADDRESS
0432*					
0433	11322	000000	BPEND	NOP	END OF BASE PAGE BUFFER
0434	11323	000000	BP#	NOP	# OF BASE PAGE LINK WORDS
0435	11324	000000	BSBAD	NOP	SEGMENT BP RELOC ADDR
0436	11325	000000	BSPAD	NOP	SEGMENT PROG RELOC ADDR
0437	11326	000000	BSSDP	NOP	INITIAL DISC RES BSS DISPLACEMENT
0438	11327	000000	BTRSC	NOP	CURRENT UTIL PROG T/S ADDRESS
0439	11330	000000	BGBND	NOP	USER PROG BOUNDARY
0440	11331	000000	BPOIT	NOP	NEG OF BP STARTING ADDRESS
0441*					
0442	11332	000000	CBASE	NOP	CURRENT BASE PROGRAM ADDRESS
0443	11333	000000	CLXXX	NOP	
0444	11334	000000	CNFLG	NOP	TEMPORARY FLAG
0445	11335	000000	COUNT	NOP	# OF BINARY SECTOR COUNTER
0446	11336	000000	CURAD	NOP	CURRENT DBUF ADDRRESS
0447	11337	000000	CURAI	NOP	CURRENT IBUF ADDRESS
0448	11340	000000	CURAL	NOP	CURRENT LBUF ADDRESS
0449	11341	000000	CURAP	NOP	CURRENT PLIST ADDRESS
0450	11342	000000	CURAT	NOP	CURRENT TBUF ADDRESS
0451*					
0452	11343	000000	DBASE	NOP	CURRENT BASE DISC ADDRESS
0453	11344	000000	DBFLG	NOP	NO DEBUG/DEBUG OPTION
0454	11345	000000	DBFL2	NOP	DEBUG IN LST FLAG
0455	11346	000000	DBLAD	NOP	CURRENT DBL ADDRESS
0456	11347	000000	DEBUGF	NOP	SPECIAL DEBUG ENTRY DEBUG FLAG
0457	11350	000000	DCNT	NOP	CURRENT DBUF COUNT
0458	11351	000000	DCRNT	NOP	CURRENT STORE DISC ADDRESS
0459	11352	000000	DISCX	NOP	WHICH DISC? FLAG
0460	11353	000000	DSKAB	NOP	STARTING ABS PROG DISC ADDRESS
0461	11354	000000	DSKAD	NOP	CURRENT DISC ADDRESS
0462	11355	000000	DSKBR	NOP	CURRENT MAIN DISC RES DISC ADDR
0463	11356	000000	DSKBS	NOP	DISC ADDR OF MAIN DISC RES BP
0464	11357	000000	DSKID	NOP	ID SEGMENT DISC ADDRESS
0465	11360	000000	DSKMN	NOP	INITIAL MAIN DISC ADDRESS
0466	11361	000000	DSKRD	NOP	DISC INPUT ADDRESS
0467	11362	000000	DSKSS	NOP	BINARY SCRATCH ADDRESS
0468	11363	000000	DSKTM	NOP	TEMPORARY DISC ADDRESS
0469	11364	000000	DSK01	NOP	
0470	11365	000000	DSK02	NOP	
0471	11366	000000	DSKUD	NOP	
0472	11367	000000	DSCNT	NOP	# SECTORS FOR CURRENT PROGRAM
0473*					
0474	11370	000000	ETFLG	NOP	END OF TAPE FLAG
0475	11371	000000	EXCNT	NOP	TEMPORARY COUNTER
0476	11372	000000	EXORD	NOP	EXTERNAL ORDINAL
0477*					

0478	11373	000000	FIOPT NOP	FILE SEARCH OPTION
0479	11374	000000	HDFLG NOP	HEADING FORMAT FLAG
0480*				
0481	11375	000000	ICNT NOP	CURRENT IBUF COUNT
0482	11376	000000	IDMBS NOP	USER MAIN ADDR FOR SEGMENT REF
0483	11377	177777	IGNOP DEC -1	LOADING/NOT LOADING FLAG
0484	11400	000000	IMAIN NOP	MAIN PROG IDENT ADDRESS
0485	11401	000000	INSCN NOP	INSTRUCTION TYPE COUNTER
0486	11402	000000	INSTR NOP	CURRENT INSTRUCTION
0487	11403	000000	LCNT NOP	CURRENT LBUF COUNT
0488	11404	000000	LDING NOP	PROGRAMS LOADING FLAG
0489	11405	000000	LOWER NOP	LOWER BP LST BOUND
0490	11406	000000	MAXAD NOP	CURRENT MAX PROG LOAD ADDRESS
0491*				
0492	11407	000000	NBPBF NOP	NEG BASE PAGE BUFFER ADDRESS
0493	11410	000000	NXFLG NOP	ENT/EXT FLAG
0494*				
0495	11411	000000	OPRND NOP	CURRENT OPERAND
0496*				
0497	11412	000000	PAGNO NOP	CURRENT PAGE NO.
0498	11413	000000	PBREL NOP	INITIAL BP RELOCA ADDR
0499	11414	000000	PCNTT NOP	PROGRAM NAME COUNTER
0500	11415	000000	PGCNT NOP	# PROGRAMS LOADED COUNTER
0501	11416	000000	PGMIN NOP	INPUT UNIT (SECONDARY)
0502	11417	000000	PLCNT NOP	CURRENT PLOST COUNT
0503	11420	000000	PLFLG NOP	TEMPORARY FLAG
0504	11421	000000	PLGTH NOP	PROGRAM LENGTH
0505	11422	000000	PLMAN NOP	CURRENT MAIN PLIST ADDRESS
0506	11423	000000	PPREL NOP	INITIAL PROG RELOC ADDR
0507	11424	000000	PRENT NOP	PRIMARY ENTRY POINT
0508	11425	000000	PTYPE NOP	CURRENT PROGRAM TYPE
0509*				
0510	11426	000000	RANAD NOP	CURRENT POWER RANGE ADDRESS
0511	11427	000000	RELAD NOP	CURRENT CORE RELOCATION ADDRESS
0512	11430	000000	REKEY NOP	INSTRUCTION TYKPE BYTE
0513	11431	000000	RIC NOP	RECORD ID CODE
0514	11432	000000	RLGTH NOP	RELOCATABLE UTIL REC LENGTH
0515*				
0516	11433	000000	SAVSS NOP	SAVE S3 FLAG
0517	11434	000000	SYM12 NOP	SYMBOL ADDRESS 1
0518	11435	000000	SYM34 NOP	SYMBOL ADDRESS 2
0519	11436	000000	SYM5 NOP	SYMBOL ADDRESS 3
0520*				
0521	11437	000000	TABS NOP	ABSOLUTE WORD
0522	11440	000000	TADR NOP	ABSOLUTER ADDRESS
0523	11441	000000	TBREL NOP	CURRENT BP RELOC ADDR
0524	11442	000000	TCNT NOP	CURRENT TBUF COUNT
0525	11443	000000	TDIR NOP	BINARY READ DIRECTION FLAG
0526	11444	000000	TEMP NOP	TEMPORARY
0527	11445	000000	TERMN NOP	TERMINATOR FLAG
0528	11446	000000	TNEXT NOP	NEXT TRACK SECTOR ADDRESS
0529	11447	000000	TTYNO NOP	TTY #
0530	11450	000000	TWORD NOP	WORD IN BUFFER ADDRESS
0531*				
0532	11451	000000	UPPER NOP	UPPER BP LST BOUND
0533	11452	000000	UTCNT NOP	# OF UTIL RECORDS LEFT (-)

PAGE 0105 #67 DOS=III LOADER - CONSTANTS & VARIABLES

0534	11453	000000	UTRSC NOP	CURRENT UTIL REC T/S ADDRESS
0535*				
0536	11454	000000	WDCNT NOP	TEMPORARY COUNTER
0537*				
0538	11455	000000	XPREL NOP	TEMPORARY COUNTER
0539*				
0540	11456	000000	UBFWA NOP	LOWER BASE PAGE BOUND
0541	11457	000000	UBLWA NOP	UPPER BASE PAGE BOUND
0542	11460	000000	UMFWA NOP	LOWER MAIN PROG BOUND
0543	11461	000000	UMLWA NOP	UPPER MAIN PROG BOUND
0544	11462	000000	DVR NOP	DEVICE DRIVER FLAG

0546	11463	000606	LISTU OCT 606	LIST OUTPUT UNIT
0547	11464	000000	ZBUF BSS 3	CURRENT PROGRAM NAME BUFFER
0548	11467	030061	ERR01 ASC 1,01	CHECKSUM ERROR
0549	11470	030062	ERR02 ASC 1,02	ILLEGAL RECORD
0550	11471	030063	ERR03 ASC 1,03	MEMORY OVERFLOW
0551	11472	030064	ERR04 ASC 1,04	LINKAGE AREA OVERFLOW
0552	11473	030065	ERR05 ASC 1,05	LOADER SYMBOL TABLE OVERFLOW
0553	11474	030066	ERR06 ASC 1,06	DUPLICATE PROGRAM NAMES
0554	11475	030067	ERR07 ASC 1,07	DUPLICATE ENTRY POINTS
0555	11476	030070	ERR08 ASC 1,08	NO TRANSFER ADDRESS PRESENT
0556	11477	030071	ERR09 ASC 1,09	RECORD OUT OF SEQUENCE
0557	11500	030460	ERR10 ASC 1,10	
0558	11501	030461	ERR11 ASC 1,11	PROGRAM IDENT TABLE OVERFLOW
0559	11502	030462	ERR12 ASC 1,12	SPECIFIED USER FILE NOT IN DIRECT
0560	11503	030463	ERR13 ASC 1,13	DUPL PROGRAM NAME
0561	11504	030464	ERR14 ASC 1,14	NON-ZERO BASE PAGE LENGTH
0562	11505	030465	ERR15 ASC 1,15	SEGMENT LOADED BEFORE MAIN,
0563	11506	030466	ERR16 ASC 1,16	PROGRAM OVERLAY
0564	11507	047117	ASCNO ASC 1,NO	
0565	11510	000331	XSSFL DEF SSFLG	ADDRESS OF SS FLAG IN BASE PAGE
0566	11511	011512R	BPBUF DEF BPAGE	ADDR OF BASE PAGE BUFFER
0567	11512		BPAGE BSS 0	BEGINNING OF BASE PAGE BUFFER

0569	00000	A	EQU 0	A REGISTER
0570	00001	B	EQU 1	B REGISTER
0571	00100	.	EQU 100B	
0572	00053	..	EQU 53B	
0573	00041	N10	EQU ..-10	DEC -10
0574	00042	N9	EQU ..-9	DEC -9
0575	00043	N8	EQU ..-8	DEC -8
0576	00044	N7	EQU ..-7	DEC -7
0577	00045	N6	EQU ..-6	DEC -6
0578	00046	N5	EQU ..-5	DEC -5
0579	00047	N4	EQU ..-4	DEC -4
0580	00050	N3	EQU ..-3	DEC -3
0581	00051	N2	EQU ..-2	DEC -2
0582	00052	N1	EQU ..-1	DEC -1
0583	00053	N0	EQU ..	DEC 0
0584	00054	.1	EQU ..+1	DEC 1
0585	00055	.2	EQU ..+2	DEC 2
0586	00056	.3	EQU ..+3	DEC 3
0587	00057	.4	EQU ..+4	DEC 4
0588	00060	.5	EQU ..+5	DEC 5
0589	00061	.6	EQU ..+6	DEC 6
0590	00062	.7	EQU ..+7	DEC 7
0591	00063	.8	EQU ..+8	DEC 8
0592	00064	.9	EQU ..+9	DEC 9
0593	00065	.10	EQU ..+10	DEC 10
0594	00066	.17	EQU ..+11	DEC 17
0595	00070	M17	EQU ..+13	OCT 17
0596	00071	M37	EQU ..+14	OCT 37
0597	00072	M77	EQU ..+15	OCT 77
0598	00073	M177	EQU ..+16	OCT 177
0599	00074	M377	EQU ..+17	OCT 377
0600	00075	M1774	EQU ..+18	OCT 177400
0601	00076	M3777	EQU ..+19	OCT 3777
0602	00077	M1777	EQU ..+20	OCT 177700
0603	00100	SMLWA	EQU ,+0	LWA OF USEABLE MEMORY
0604	00101	JBINS	EQU ,+1	START T/S OF JBIN AREA
0605	00102	JBINC	EQU ,+2	CURRENT T/S OF JBIN AREA
0606	00111	BATCH	EQU ,+9	L.U. OF BATCH INPUT DEVICE
0607	00112	SYSTY	EQU ,+10	L.U. OF SYSTEM TELETYPE
0608	00116	SECTR	EQU ,+14	# OF SECTORS/TRACK
0609	00147	EXPG7	EQU ,+39	
0610	00154	DISCO	EQU ,+44	LAST TRACK ON DISC
0611	00155	SYSSC	EQU ,+45	
0612	00157	UDNTS	EQU ,+47	NEXT USER AVAILABLE T/S
0613	00160	SYNTS	EQU ,+48	NEXT SYSTEM AVAILABLE T/S
0614	00161	CUDSC	EQU ,+49	CURRENT USER DISC SUBCHANNEL
0615	00174	TSONE	EQU ,+60	NEXT T/S AFTER LAST REQUEST
0616	00200	DISCL	EQU ,+64	#10 IF USER AREA NOT ON SYS DISC
0617	00254	SMFWA	EQU ,+108	FWA OF USER MAIN AREA
0618	00255	SBFWA	EQU ,+109	FWA OF USER BASE AREA
0619	00256	SBLWA	EQU ,+110	LWA OF USER BASE AREA
0620	00313	MSECT	EQU ,+139	NEG # OF SECTORS/TRACK
0621	00324	DSCLB	EQU ,+148	RELOCATABLE LIBRARY T/S
0622	00325	DSCL#	EQU ,+149	# OF RELOCATABLE LIBRARY SECTORS
0623	00331	SSFLG	EQU ,+153	
0624		END	LOADR	

PAGE 0107 #07 DOS-III LOADER - CONSTANTS & VARIABLES

** NO ERRORS*

#LIB	00326/02	00191/02	00322/02				
#SECT	00505/01	00483/01	00487/01				
#WRDS	00238/05	00040/03	00043/03	00069/03	00098/03	00174/03	00179/03
	00232/05						
#SDBP1	00335/07						
#SDBP2	00339/07						
#SMEMR	00341/07						
.	00571/07	00603/07	00604/07	00605/07	00606/07	00607/07	00608/07
	00609/07	00610/07	00611/07	00612/07	00613/07	00614/07	00615/07
	00616/07	00617/07	00618/07	00619/07	00620/07	00621/07	00622/07
	00623/07						
..	00572/07	00573/07	00574/07	00575/07	00576/07	00577/07	00578/07
	00579/07	00580/07	00581/07	00582/07	00583/07	00584/07	00585/07
	00586/07	00587/07	00588/07	00589/07	00590/07	00591/07	00592/07
	00593/07	00594/07	00595/07	00596/07	00597/07	00598/07	00599/07
	00600/07	00601/07	00602/07				
.1	00584/07	00154/01	00291/01	00339/01	00344/01	00421/01	00562/01
	00604/01	00174/02	00202/02	00246/03	00112/04	00078/05	00643/06
	00649/06						
.10	00593/07	00548/01	00432/03	00599/03	00609/03	00629/05	
.11	00313/07	00420/03					
.12	00314/07	00341/01					
.127	00319/07	00591/05	00602/05				
.128	00362/07	00480/01	00565/01	00042/03	00097/03	00154/03	00173/03
	00178/03	00274/03	00597/03	00607/03	00170/05	00270/05	00459/05
	00294/06	00299/06					
.14	00315/07	00631/03	00350/05				
.16	00316/07	00290/01	00391/04	00497/06			
.17	00594/07	00216/01	00601/04				
.18	00317/07	00364/01	00364/03	00465/03	00627/03		
.2	00585/07	00308/01	00338/01	00433/01	00466/01	00511/01	00521/01
	00590/01	00608/01	00150/02	00172/02	00218/02	00411/03	00046/04
	00151/04	00077/05	00116/05	00154/05	00509/06	00642/06	00706/06
.3	00586/07	00468/01	00610/01	00220/02	00501/02	00503/02	00660/02
	00722/02	00053/03	00057/03	00095/03	00227/03	00348/03	00358/03
	00423/03	00462/03	00479/03	00617/03	00043/04	00153/04	00266/04

	00363/04	00529/04	00606/04	00024/06	00515/06		
.36	00318/07	00469/03	00102/04	00260/04			
.4	00587/07	00470/01	00478/01	00612/01	00222/02	00350/03	00155/04
	00185/04	00483/04	00088/05	00573/05	00574/05	00618/05	
.5	00588/07	00375/01	00458/01	00472/01	00614/01	00054/02	00224/02
	00469/02	00669/02	00697/02	00704/02	00352/03	00372/03	00426/03
	00527/03	00541/03	00063/04	00157/04	00467/04	00485/04	00263/05
	00353/05	00585/05	00597/05				
.6	00589/07	00507/02	00675/03	00116/04	00129/04	00422/04	00487/04
	00287/05						
.7	00590/07	00553/01	00161/02	00239/03	00111/04	00131/04	00150/04
	00169/04	00424/04	00482/04	00289/05	00043/07		
.8	00591/07	00645/06					
.9	00592/07	00417/03	00126/04				
A	00569/07	00109/04	00117/04	00127/04	00148/04	00214/04	00366/04
	00490/04	00532/04	00556/04	00609/04	00063/05	00325/05	00374/05
	00387/05	00390/05	00698/05	00690/06	00048/07	00141/07	00147/07
	00158/07	00172/07					
AABUF	00465/05	00534/01	00128/05	00138/05	00445/05		
ABBUF	00466/05	00131/05	00136/05	00448/05			
ABFLG	00438/05	00538/01	00129/05	00135/05	00424/05	00446/05	00452/05
ABORT	00497/06	00378/01	00472/02	00672/02	00382/03	00060/05	00571/06
	00578/06						
ABRTX	00496/06	00618/01	00621/01	00227/02	00521/02	00665/02	00409/03
	00188/04	00396/05	00749/05	00009/06			
ABSID	00430/07	00551/03	00578/03				
ABUF	00467/05	00465/05					
ADBUF	00364/07	00045/03	00056/03	00065/03	00475/03	00594/03	00598/03
	00603/03	00660/03	00665/03	00770/05	00060/06	00064/06	00084/06
	00100/06	00104/06	00117/06				
AEXEC	00345/07	00259/01	00261/01	00263/01	00101/02	00105/02	00111/02
AIBUF	00365/07	00389/03	00395/03	00446/03	00608/03	00644/03	00715/05
	00717/05						
ALBUF	00366/07	00580/01	00029/02	00148/03	00151/03		
ALINK	00120/07	00119/04	00123/07	00183/07	00281/07	00285/07	00289/07

ALLOC	00382/05	00400/04	00331/05	00394/05			
AMAD	00431/07	00431/01	00434/01	00438/01	00440/01	00444/01	00448/01
	00451/01	00452/01	00453/01	00019/04	00023/04	00025/04	00026/04
	00029/04	00030/04	00036/04	00352/06	00356/06		
AME12	00369/07	00096/04	00252/04				
AME18	00370/07	00258/04					
AMEM5	00367/07	00093/04	00249/04	00389/04			
AMEM8	00368/07	00255/04					
AMLST	00371/07	00018/04	00103/04	00261/04	00392/04	00350/06	
ANEW	00103/05	00100/05					
ASCNO	00564/07	00169/01	00668/03				
B	00570/07	00136/01	00141/01	00147/01	00150/01	00153/01	00206/01
	00223/01	00435/01	00441/01	00446/01	00554/01	00591/01	00598/01
	00162/02	00166/02	00463/02	00563/02	00568/02	00058/03	00240/03
	00256/03	00412/03	00601/03	00611/03	00077/04	00083/04	00139/04
	00141/04	00144/04	00194/04	00211/04	00226/04	00237/04	00240/04
	00243/04	00246/04	00295/04	00298/04	00304/04	00317/04	00325/04
	00353/04	00356/04	00431/04	00446/04	00453/04	00523/04	00569/04
	00576/04	00578/04	00583/04	00589/04	00661/04	00187/05	00191/05
	00193/05	00249/05	00519/05	00522/05	00543/05	00552/05	00576/05
	00579/05	00775/05	00013/06	00015/06	00047/06	00069/06	00078/06
	00168/06	00175/06	00221/06	00226/06	00260/06	00267/06	00280/06
	00316/06	00354/06	00404/06	00424/06	00431/06	00073/07	00136/07
	00217/07						
B1777	00585/06	00552/06					
BATCH	00606/07	00328/01	00332/01	00568/06	00575/06	00631/06	00635/06
BBADD	00644/06	00628/06					
BBUF	00468/05	00466/05					
BD005	00631/03	00626/03					
BD010	00634/03	00630/03					
BDEND	00623/03	00492/03					
BOLD	00491/03	00495/03	00622/03				
BGBND	00439/07	00452/03					
BGSPL	00069/04	00064/04					
BIDNT	00417/07	00199/01	00483/03	00018/05			

BLANK 00330/07 00035/04 00386/04 00684/06

BLAST 00328/07 00378/04

BLCK# 00184/03 00096/03 00140/03 00153/03 00158/03 00172/03 00254/03

BLCK1 00169/03 00144/03

BLCK2 00146/03 00176/03

*BLCK3 00164/03

BLCK5 00161/03 00166/03

BLCK6 00178/03 00207/02 00324/02 00171/03

BLIN0 00279/07 00236/07 00273/07 00292/07

BLIN1 00285/07 00291/07

*BLIN2 00237/07

BLIN3 00240/07 00721/03 00228/07 00253/07

BLINK 00263/07 00222/04 00200/07 00206/07 00209/07 00213/07 00223/07
00265/07 00272/07 00277/07

BLKAD 00183/03 00414/01 00196/02 00301/02 00137/03 00147/03 00162/03
00164/03

BLNKS 00329/07 00353/06 00392/06

BLOCK 00135/03 00416/01 00199/02 00167/03 00309/03

BLST 00407/07 00191/01 00676/04

BMES 00586/06 00545/06

BMES1 00588/06 00556/06

BNDHD 00272/04 00236/04

BP# 00434/07 00187/01 00201/01

BPAOD 00377/05 00370/05 00375/05

BPAGE 00567/07 00566/07

BPALC 00330/05 00319/05

BPBE 00566/06 00548/06 00554/06

BPBUF 00566/07 00179/01 00188/01 00193/01 00204/01 00555/04 00659/04
00323/05 00372/05 00385/05 00697/05 00264/06 00282/06

BPEND 00433/07 00189/01

BPN 00136/04 00124/04 00133/04

BPOIT 00440/07 00185/01 00554/04 00658/04 00322/05 00371/05 00384/05
00696/05 00265/06 00283/06

BPOUT 00278/06 00522/03 00564/03 00587/03 00286/06 00297/06

BPSCN 00310/05 00371/04 00574/04 00329/05 00332/05 00186/07

BPSRC 00316/05 00327/05

BPSTR 00030/06 00174/01 00651/03 00658/03 00672/03 00034/06

BPSYO 00287/06 00301/06

BS1 00612/03 00590/03 00605/03

BSBAD 00435/07 00526/03 00563/03 00565/03 00574/03 00585/03 00615/03
00598/05

BSEND 00581/03 00534/03

BSLD 00533/03 00539/03 00543/03 00549/03 00580/03

BSPAD 00436/07 00524/03 00576/03 00589/03 00586/05

BSSDP 00437/07 00470/04 00587/05

•BTRSC 00438/07

BUFCL 00245/05 00536/01 00163/05 00253/05 00719/05 00061/06 00101/06

BUMP 00299/03 00249/02 00250/02 00253/02 00256/02 00259/02 00286/02
00418/02 00419/02 00422/02 00425/02 00429/02 00432/02 00435/02
00438/02 00558/02 00559/02 00616/02 00617/02 00621/02 00624/02
00628/02 00740/02 00235/03 00307/03 00310/03 00330/03

BYTE 00599/04 00488/04

CBASE 00442/07 00472/04 00750/05

CBLA 00185/03 00138/03 00261/03 00269/03

CBUF 00458/05 00449/05 00482/05

CDEC 00702/05 00667/05

CF10 00177/02 00173/02

CFILE 00172/02 00164/02

CHKER 00623/01 00600/01

•CHKOK 00237/03

CHRDB 00343/07 00154/06 00158/06 00163/06

CHRDE 00337/07 00083/02 00085/02 00087/02 00286/04 00290/04 00296/04
00626/04 00630/04 00634/04 00641/04 00645/04 00650/04 00142/06
00146/06 00151/06

CIDNT 00420/07 00484/03 00369/06 00376/06

CL 00162/05 00157/05

CLBPS 00257/06 00245/06

CLFLG 00234/04 00276/04 00279/04

CLID3 00262/05 00619/03 00268/04 00275/05 00279/05

CLINK 00303/07 00187/07

CLIST 00349/06 00012/04 00235/04 00377/04 00225/06 00228/06 00358/06

CLORD 00315/02 00320/02

CLR0 00356/03 00379/03

CLRID 00341/03 00355/03 00369/03 00376/03

CLRLT 00238/06 00567/03 00616/03 00005/04 00262/06 00271/06

CLRNX 00244/06 00249/06 00253/06 00256/06

CLRTK 00038/06 00769/05 00086/06

CLSRC 00146/04 00449/04 00503/04

CLST 00199/04 00204/04

CLSUT 00296/05 00301/05 00303/05

CLXXX 00443/07 00277/04 00280/04 00134/06 00135/06

CMES 00389/01 00340/01

CMESS 00679/03 00632/03

CMND 00437/05 00419/05 00457/05

CNFLG 00444/07 00645/01 00018/02 00045/02

COMAD 00395/07 00453/03

COMMA 00352/07 00411/06 00686/06

CON 00040/07 00033/07 00036/07 00050/07

CONV 00024/07 00727/06 00006/07 00064/07

CONV1	00030/07	00038/07					
CONV2	00052/07	00042/07					
CONV3	00081/07	00053/07					
CONVD	00308/06	00094/04	00097/04	00250/04	00253/04	00256/04	00259/04
	00390/04	00332/06					
COREA	00231/05	00205/05	00218/05				
COUNT	00445/07	00282/01	00302/01	00304/01	00366/01	00370/01	00194/02
	00142/03	00146/03	00169/03				
CPADD	00307/07	00182/07	00194/07	00249/07			
CPBAS	00304/07	00180/07	00201/07	00247/07	00282/07		
CPL	00300/07	00151/01	00131/07				
CPLOC	00305/07	00179/07	00203/07	00216/07	00221/07	00246/07	00258/07
	00267/07	00275/07					
CPLST	00076/04	00068/04					
CPMAX	00306/07	00174/07	00218/07	00251/07			
CPXXX	00082/04	00075/04					
CROSS	00226/07	00167/04	00197/07				
CUDSC	00614/07	00240/01	00653/03				
CURAD	00446/07	00535/01	00036/02	00041/02	00476/03	00139/05	00162/05
CURAI	00447/07	00390/03	00397/03	00410/03	00418/03	00424/03	00709/05
	00710/05	00718/05					
CURAL	00448/07	00432/01	00581/01	00588/01	00589/01	00596/01	00239/02
	00251/02	00254/02	00257/02	00420/02	00423/02	00426/02	00430/02
	00433/02	00436/02	00439/02	00554/02	00560/02	00611/02	00619/02
	00622/02	00625/02	00025/03	00152/03	00215/03	00237/03	00301/03
	00107/04	00115/04	00125/04	00136/04	00140/04	00142/04	00147/04
	00166/04	00195/04	00331/04	00334/04	00337/04	00367/04	00459/04
	00474/04	00491/04	00508/04	00520/04	00525/04	00533/04	00599/04
	00605/04	00613/04	00667/04	00403/05	00483/05		
CURAP	00449/07	00486/03	00529/03	00621/03	00308/04	00311/04	00267/05
	00273/05	00276/05	00177/06	00179/06			
CURAT	00450/07	00284/06	00289/06	00298/06	00300/06	00309/06	00321/06
	00322/06	00325/06	00327/06	00328/06	00329/06		
D57	00360/07	00231/03					
DADRR	00334/07	00182/06					

DATAB 00402/07 00346/01 00355/01 00651/06
DATB 00346/01 00352/01 00660/06
DBASE 00452/07 00757/05 00123/06 00213/06
DBFL2 00454/07 00076/02 00095/02
DBFLG 00453/07 00148/01 00073/02 00274/04
DBL 00552/02 00497/01 00008/02 00233/02 00565/02 00571/02
DBL0 00473/04 00462/04
DBL1 00474/04 00507/04
DBL2 00479/04 00506/04
DBL3 00492/04 00513/04 00524/04 00587/04 00590/04 00615/04
DBL4 00508/04 00484/04
DBL5 00512/04 00518/04
DBL6 00525/04 00486/04
DBL7 00583/04 00572/04
DBL8 00586/04 00582/04
DBL9 00589/04 00563/04
DBLAD 00455/07 00460/04 00492/04 00504/04 00566/04
DBLR 00008/02 00611/01
DBSET 00402/05 00146/04 00171/04 00172/04 00329/04 00330/04 00333/04
00336/04 00339/04 00394/04 00457/04 00458/04 00473/04 00478/04
00500/04 00527/04 00612/04 00617/04 00405/05 00407/05
DBUF 00398/07 00048/03 00049/03 00055/03 00062/03 00663/03 00364/07
DEBUGF 00456/07 00653/04 00002/06 00011/06
DCNT 00457/07 00533/01 00037/02 00048/02 00474/03 00149/05
DCRNT 00458/07 00764/05 00768/05 00044/06 00048/06 00070/06 00083/06
00116/06 00124/06
DDBLR 00453/04 00154/04
DDOUT 00126/05 00039/02 00051/02 00174/05
DDRV 00378/03 00351/03

DEBUG 00624/04 00197/04 00628/04 00632/04 00636/04 00670/04
DEI10 00283/04 00288/04 00292/04 00300/04
DEIDG 00274/04 00228/04
DEN10 00196/04 00182/04 00186/04
DEN20 00197/04 00176/04
DEN30 00191/04 00179/04
DENDR 00163/04 00158/04
DENTR 00322/04 00152/04
DEXTR 00323/04 00156/04
DFPBF 00374/07 00376/01 00399/06
DFSTS 00373/07 00657/03
DFUTS 00372/07 00650/03
DINIT 00121/06 00503/03 00554/03 00128/06
DISCL 00616/07 00386/03 00659/03 00664/03
DISCO 00610/07 00242/01
DISCX 00459/07 00309/01 00512/01 00151/02 00484/02 00047/04 00230/05
00415/05 00025/06
DISKA 00185/05 00145/05 00192/05 00196/05 00537/05 00760/05 00214/06
DISKI 00204/05 00046/03 00149/03 00396/03 00595/03 00661/03 00209/05
00085/06
DISKO 00217/05 00066/03 00447/03 00604/03 00645/03 00666/03 00222/05
00716/05 00065/06 00105/06 00118/06 00290/06
DLIN1 00213/07 00205/07 00208/07
DLIN2 00206/07 00211/07
DLINK 00185/07 00330/05 00189/07 00192/07 00215/07 00219/07 00241/07
DLU 00435/05 00155/05 00416/05 00501/05
DONE 00003/07 00713/06 00718/06 00724/06 00005/07
DONE1 00006/07 00011/07
DOSHD 00347/07 00466/03
DPWRS 00384/07 00312/06

DRCH0 00403/03 00429/03
DRCH1 00410/03 00402/03
DRCH5 00423/03 00416/03
DRCH6 00428/03 00422/03
DRCHA 00408/03 00226/01 00063/02 00724/05 00043/06
DRCHK 00397/03 00430/03
DRCLR 00445/03 00439/03
DRKEY 00110/05 00467/03 00471/03 00104/04 00262/04 00393/04 00101/05
00120/05
DRSCH 00389/03 00407/03
DRWOW 00443/05 00429/05 00463/05 00489/05
DSCAN 00638/04 00643/04 00647/04 00652/04
DSCL# 00622/07 00189/02
DSCLB 00621/07 00195/02
DSCNT 00472/07 00644/01 00056/02 00149/02 00161/03 00160/05 00423/05
00487/05
DSK01 00469/07 00050/06 00077/06 00095/06
DSK02 00470/07 00055/06 00081/06 00099/06
DSKAB 00460/07 00248/01 00477/03
DSKAD 00461/07 00217/01 00220/01 00231/01 00235/01 00277/01 00367/01
00412/01 00515/01 00545/01 00630/01 00647/01 00062/02 00478/03
00518/03 00581/03 00584/03 00613/03 00649/03 00656/03 00058/04
00141/05 00144/05 00146/05 00122/06 00217/06 00288/06 00292/06
DSKBR 00462/07 00582/03 00612/03
DSKBS 00463/07 00519/03 00583/03
DSKID 00464/07 00388/03 00394/03 00404/03 00445/03 00510/03 00592/03
00641/03 00714/05 00721/05
DSKMN 00465/07 00061/04 00581/05
DSKNM 00530/02 00413/01 00001/02 00197/02 00300/02 00490/02 00265/03
DSKRD 00466/07 00413/05 00426/05 00475/05 00477/05
DSKSS 00467/07 00237/01 00514/01

DSKTM 00468/07 00246/01 00041/06
DSKUD 00471/07 00063/06 00066/06 00068/06 00107/06
DTRTM 00066/02 00230/01 00061/02
DUP 00667/02 00657/02
DVR 00544/07 00378/03 00184/04 00572/05 00617/05
DVR1 00662/05 00669/05 00672/05 00676/05 00689/05 00694/05
DVR2 00678/05 00663/05
DVRAD 00651/05 00619/05
DVRS 00685/05 00670/05 00674/05 00699/05
DVRTN 00625/05 00682/05
DWRIT 00028/02 00005/02 00019/02 00064/02
EBUF 00169/05 00132/05
EN,XT 00668/02 00502/01 00013/02 00743/02
END 00022/03 00475/01 00021/02 00031/03
END1 00322/02 00316/02
ENDDR 00431/03 00399/03
ENDR 00016/02 00615/01
ENEX1 00341/04 00346/04 00350/04 00358/04
ENEX2 00274/02 00261/02
ENEX3 00282/02 00267/02
ENEX4 00290/02 00276/02
ENTID 00305/04 00230/04
ENTR 00011/02 00609/01
ENTXX 00338/02 00343/02 00347/02 00355/02
ENXT1 00738/02 00676/02 00706/02 00714/02 00734/02
ENXX1 00271/02 00296/02
EPLST 00231/04 00314/04
ERR01 00548/07 00623/01

ERR02	00549/07	00617/01					
ERR03	00550/07	00748/05					
ERR04	00551/07	00395/05					
ERR05	00552/07	00495/06					
ERR06	00553/07	00370/03					
ERR07	00554/07	00667/02					
ERR08	00555/07	00187/04					
ERR09	00556/07	00620/01					
ERR10	00557/07	00408/03					
ERR11	00558/07	00058/05					
ERR12	00559/07	00373/01					
ERR13	00560/07	00467/02					
ERR14	00561/07	00520/02					
ERR15	00562/07	00664/02					
ERR16	00563/07	00008/06					
ERR17	00229/02	00226/02					
ERR18	00583/06	00566/06					
ERR19	00584/06	00573/06					
ERROR	00086/05	00374/01	00624/01	00468/02	00668/02	00371/03	00059/05
	00091/05	00496/06	00567/06	00574/06			
ETFLG	00474/07	00558/01	00572/01	00578/01			
EXCNT	00475/07	00097/02	00133/02	00137/02	00148/02	00153/02	00312/02
	00614/02	00741/02	00328/04	00447/04	00456/04	00501/04	00340/05
	00347/05	00356/05	00358/05	00423/06	00430/06		
EXEC	00004/01	00214/01	00288/01	00336/01	00342/01	00362/01	00551/01
	00560/01	00159/02	00362/03	00460/03	00673/03	00075/05	00114/05
	00165/05	00227/05	00454/05	00498/05	00031/06	00513/06	00640/06
	00647/06						
EXEND	00447/04	00395/04	00405/04	00410/04	00418/04	00427/04	00436/04
	00440/04	00443/04					
EXNND	00284/02	00270/02	00273/02	00292/02	00299/02		

EXORD 00476/07 00510/04 00516/04 00540/04 00546/04
EXPG7 00609/07 00178/01
EXT1 00702/02 00651/02
EXT2 00396/04 00361/04
EXTR 00012/02 00613/01
FETCH 00213/03 00420/01 00457/01 00465/01 00585/01 00201/02 00217/02
00248/03 00267/03 00279/03
FIOPT 00478/07 00321/01 00177/02
FIRST 00177/05 00539/01 00151/05 00156/05
FLLA 00256/07 00232/07
GENID 00559/05 00512/03 00557/03 00627/05 00631/05 00637/05 00642/05
GET10 00402/06 00436/06
GET15 00439/06 00418/06
GET20 00440/06 00412/06 00414/06 00416/06
GETD 00333/06 00319/06 00323/06 00326/06 00345/06
GETL 00041/02 00038/02
GETP 00391/06 00358/01 00398/06 00438/06 00441/06 00442/06 00461/06
00482/06
GOCHK 00159/02 00116/02 00176/02
GTS 00517/05 00142/05 00206/05 00219/05 00427/05 00478/05 00527/05
PHABSS 00470/04
HDFLG 00479/07 00553/03 00692/03 00020/04 00055/04 00174/04
I.O 00224/05 00208/05 00221/05 00235/05
IBUF 00399/07 00365/07
ICNT 00481/07 00392/03 00419/03 00421/03 00425/03 00427/03 00431/03
00437/03 00508/03 00625/05 00628/05 00634/05 00711/05 00726/05
ID1 00422/07 00271/02 00453/02 00470/02 00477/02 00674/02 00696/02
00708/02 00360/03 00373/03 00501/03 00535/03 00024/04 00076/04
00487/04 00033/05 00299/05 00566/05 00141/06 00153/06 00170/06
00373/06 00670/06 00701/06 00702/06 00707/06 00710/06 00026/07
00030/07 00034/07 00037/07
ID2 00423/07 00457/02 00479/02 00028/04 00035/05 00568/05 00651/05

00145/06 00157/06 00028/07 00045/07 00049/07

ID3 00424/07 00461/02 00481/02 00343/03 00345/03 00493/03 00496/03
00032/04 00312/04 00315/04 00413/04 00037/05 00292/05 00294/05
00577/05 00655/05 00149/06 00161/06 00166/06 00629/06 00059/07
00067/07

ID4 00425/07 00488/02 00497/03 00044/04 00039/05 00062/07 00071/07
00077/07

ID5 00426/07 00491/02 00105/04 00041/05 00659/06 00055/07 00063/07
00081/07

ID6 00427/07 00052/02 00508/02 00658/02 00694/02 00702/02 00718/02
00027/03 00028/03 00346/03 00356/03 00359/03 00537/03 00411/04
00465/04 00043/05 00283/05 00570/05 00377/06 00661/06 00676/06
00681/06

ID7 00428/07 00485/01 00491/01 00057/02 00511/02 00736/02 00544/03
00014/04 00048/05 00663/06 00675/06 00682/06

IDCLR 00380/03 00342/03

IDEC 00701/05 00665/05

IDMBS 00482/07 00502/03 00547/03

IDSCN 00368/06 00491/03 00372/06 00382/06 00383/06 00384/06

IDX 00028/05 00257/01 00445/02 00716/02 00730/02 00341/03 00533/03
00309/04 00409/04 00417/04 00032/05 00054/05 00062/05 00281/05
00564/05 00139/06 00371/06

IGNOR 00483/07 00188/02 00212/02 00231/02 00243/02 00262/02 00290/02
00303/02 00306/02 00311/02 00448/02

IMAIN 00484/07 00709/02 00727/02 00536/03 00562/05 00374/06

INCOR 00606/03 00593/03

INCRA 00335/06 00340/06

INDSK 00090/06 00108/06 00125/06

INIDX 00017/05 00256/01 00444/02 00340/03 00532/03 00020/05 00138/06

INLST 00675/04 00253/01 00079/02 00098/02 00140/02 00205/02 00314/02
00337/02 00630/02 00198/04 00282/04 00340/04 00511/04 00541/04
00637/04 00678/04 00295/05 00341/05 00661/05 00243/06

INNAM 00412/05 00106/04 00433/05

INREL 00473/05 00406/05 00431/05 00491/05

INSCN 00485/07 00477/04 00505/04

INSTR	00486/07	00526/04	00528/04	00535/04	00558/04	00564/04	00586/04
IP	00674/05	00666/05					
JBINC	00605/07	00278/01	00280/01	00283/01			
JBINS	00604/07	00274/01					
JBN10	00288/01	00305/01					
JBN20	00301/01	00296/01					
JBN30	00306/01	00298/01					
L001	00301/07	00710/03	00713/03	00724/03	00127/07	00230/07	
L01	00302/07	00723/03	00007/04	00049/04	00059/04	00099/04	00163/04
	00220/04	00374/04	00463/04	00495/04	00121/07	00190/07	00226/07
LAB10	00750/05	00747/05					
LAB19	00082/06	00052/06	00057/06	00059/06			
LAB20	00770/05	00765/05					
LAB25	00010/06	00004/06					
LAB30	00012/06	00001/06	00007/06				
LAB40	00014/06	00774/05					
LABD0	00740/05	00219/04	00499/04	00663/04	00019/06	00021/06	00287/07
LBERR	00226/02	00204/02					
LBUF	00400/07	00564/01	00031/02	00366/07			
LCNT	00487/07	00583/01	00034/02	00043/02	00156/03	00251/03	00273/03
	00306/03	00143/04	00145/04	00404/05	00485/05		
LD1	00720/03	00715/03					
LD2	00006/04	00002/04					
LD3	00710/03	00261/07					
LD4	00055/04	00051/04					
LD5	00058/04	00054/04					
LD6	00105/04	00101/04					
LD7	00168/04	00165/04					
LD8	00500/04	00497/04	00610/04				

LDABS 00451/03 00434/03
LDBLR 00231/02 00221/02
LDING 00488/07 00490/03 00055/05
LDRIN 00560/01 00574/01 00006/02 00022/02
LENDR 00306/02 00225/02
LENTR 00236/02 00219/02
LEXTR 00237/02 00223/02
LIB1 00201/02 00323/02
LIBRY 00643/05 00488/03 00517/03 00561/03 00122/04 00135/04 00671/05
00680/05
LIBS 00183/02 00152/02 00184/02
LIBUT 00428/04 00423/04 00425/04
LINKC 00575/04 00224/07 00303/07
PLINNM 00218/02
LIRD 00199/02 00304/02
LISKP 00216/02 00214/02 00234/02 00248/02 00289/02
LISTU 00546/07 00145/01 00456/03 00117/05
LMAIN 00338/03 00139/02 00142/02 00158/02 00175/02
LO01 00178/01 00170/01
LOAD 00690/03 00505/03 00698/03
LOAD1 00168/01 00562/06
LOADL 00098/02 00122/02 00132/02 00136/02
LOADN 00709/03 00318/04
LOADR 00136/01 00005/01 00401/07 00404/07 00405/07 00624/07
LOADS 00706/03 00556/03 00697/03 00267/04 00269/04
LOADX 00723/03 00167/07 00238/07
LODLB 00073/02 00519/01 00522/01 00556/01
LODLL 00137/02 00100/02 00155/02
LODLX 00123/02 00104/02 00108/02 00113/02

LOVER 00495/06 00011/05

LOWER 00489/07 00312/05 00316/05 00321/05 00324/05 00240/06 00247/06
00257/06

LPARN 00353/07 00413/06 00458/06

LSCAN 00336/02 00260/02 00339/02 00356/02 00357/02 00193/06

LST1 00411/07 00260/01 00084/02 00102/02 00341/02 00363/02 00634/02
00670/02 00681/02 00285/04 00344/04 00380/04 00640/04 00692/04
00354/05 00664/05 00672/06 00698/06 00711/06

LST10 00342/05 00346/05 00357/05

LST2 00412/07 00262/01 00086/02 00106/02 00345/02 00365/02 00638/02
00683/02 00289/04 00348/04 00382/04 00644/04 00694/04 00686/05
00673/06 00696/06 00717/06 00721/06

LST20 00358/05 00343/05

LST3 00413/07 00264/01 00089/02 00109/02 00119/02 00121/02 00125/02
00127/02 00283/02 00317/02 00319/02 00348/02 00368/02 00641/02
00685/02 00201/04 00203/04 00293/04 00354/04 00384/04 00397/04
00514/04 00544/04 00648/04 00696/04 00690/05 00195/06 00197/06

LST4 00414/07 00266/01 00091/02 00115/02 00128/02 00135/02 00143/02
00288/02 00268/02 00272/02 00279/02 00297/02 00653/02 00675/02
00780/02 00711/02 00735/02 00302/04 00402/04 00435/04 00444/04
00549/04 00654/04 00002/05 00298/05 00344/05 00198/06 00200/06

LST5 00415/07 00267/01 00092/02 00118/02 00124/02 00281/02 00295/02
00687/02 00369/04 00372/04 00398/04 00401/04 00519/04 00553/04
00657/04 00004/05 00302/05 00695/05 00202/06 00246/06 00250/06
00255/06

LSTCR 00205/04 00200/04

LSTEX 00338/05 00156/02 00360/05 00361/05 00362/05

LSTX 00687/04 00254/01 00080/02 00099/02 00141/02 00206/02 00315/02
00338/02 00631/02 00199/04 00283/04 00341/04 00512/04 00542/04
00638/04 00691/04 00010/05 00296/05 00342/05 00662/05 00244/06

LTEMP 00067/02 00030/02 00035/02 00042/02

LTERM 00638/03 00500/06

LTERW 00673/03 00669/03

LTERX 00659/03 00655/03

LTERZ 00667/03 00643/03 00648/03

LTOD 00035/02 00044/02

LWH1	00296/07	00495/02	00036/03	00064/03	00712/03	00166/04	00264/07
LWH2	00297/07	00569/02	00047/03	00169/07	00176/07	00234/07	00259/07
	00270/07	00290/07					
LWH3	00298/07	00719/03	00003/04	00010/04	00089/04	00095/04	
LWH4	00299/07	00717/03	00006/04	00087/04	00092/04	00471/04	
LWR	00062/07	00084/07					
M0760	00325/07	00193/02	00567/04	00570/04	00181/07	00248/07	
M1100	00322/07	00458/03					
•M17	00595/07						
M1740	00326/07	00521/04	00559/04				
M177	00598/07	00053/02	00498/02	00659/02	00695/02	00703/02	00721/02
	00051/03	00219/03	00347/03	00413/03	00540/03	00128/04	00421/04
	00466/04	00286/05	00571/05	00652/05	00657/05	00227/06	00379/06
	00679/06	00032/07	00035/07				
M1774	00600/07	00447/01	00088/02	00110/02	00120/02	00126/02	00318/02
	00349/02	00352/02	00367/02	00427/02	00462/02	00626/02	00642/02
	00344/03	00357/03	00033/04	00202/04	00294/04	00297/04	00352/04
	00355/04	00385/04	00649/04	00194/05	00293/05	00578/05	00691/05
	00045/06	00074/06	00092/06	00150/06	00162/06	00196/06	00429/06
	00472/06						
•M1777	00602/07						
M200	00356/07	00575/05					
M2000	00324/07	00580/04	00585/04	00250/07			
M240	00569/01	00567/01					
M300	00357/07	00138/01	00169/02				
M37	00596/07	00240/02	00612/02	00326/04			
M377	00599/07	00281/01	00285/01	00405/03	00138/04	00509/04	00515/04
	00537/04	00545/04	00188/05	00520/05	00524/05	00544/05	00722/05
	00040/06	00049/06	00079/06	00097/06	00407/06	00427/06	
M3777	00601/07	00243/01					
M400	00358/07	00325/01	00330/01	00195/05	00633/06		
M60	00355/07	00344/06					
M600	00320/07	00144/01					

M700 00321/07 00511/06
M7600 00359/07 00592/05 00603/05 00222/06
M77 00597/07 00520/01 00555/02 00487/03 00454/04 00506/06
MAXAD 00490/07 00017/06 00020/06 00127/06
MAXC 00308/07 00512/02 00561/02 00570/02 00283/07 00286/07 00288/07
MAXSG 00683/03 00507/03 00569/03 00573/03 00588/03 00600/03 00610/03
MAXWK 00061/02 00047/02 00055/02
MESAB 00677/03 00498/06
MESLD 00637/01 00549/01
MESSE 00092/05 00087/05 00089/05
MESUX 00363/05 00351/05
MLIST 00403/07 00238/04 00241/04 00244/04 00247/04 00379/04 00381/04
00383/04 00387/04 00367/07 00368/07 00369/07 00370/07 00371/07
MSECT 00620/07
MSIGN 00327/07 00486/02 00303/04 00445/04 00522/04 00561/04 00581/04
00665/04
N0 00583/07
N1 00582/07 00144/02 00209/02 00252/03 00400/03 00456/05
N10 00573/07 00464/03
N128 00361/07 00532/01 00582/01 00049/02 00253/03 00276/03 00391/03
00438/03 00473/03 00065/04 00072/04 00223/04 00428/04 00148/05
00246/05 00484/05 00026/05 00636/05 00725/05
N13 00510/05 00500/05
N18 00312/07 00351/06
N19 00311/07 00033/06
N2 00581/07 00224/01 00167/05 00220/05 00549/05 00178/06 00317/06
N3 00580/07
N4 00579/07 00180/06
N5 00578/07 00476/04
N6 00577/07 00197/01 00242/03 00414/03

N7	00576/07	00052/05							
N72	00310/07	00347/01	00652/06						
N8	00575/07	00693/06							
N9	00574/07	00500/02							
NAENT	00349/07	00470/03							
NAM	00414/02	00462/01	00003/02	00215/02	00519/02				
NAM12	00523/02	00421/02	00452/02	00476/02					
NAM34	00524/02	00424/02	00456/02	00478/02					
NAM5	00525/02	00428/02	00460/02	00480/02					
NAMR	00640/01	00605/01							
NB60	00582/06	00689/06							
NBPBF	00492/07	00196/01	00328/05						
NBPL	00527/02	00434/02	00517/02						
NBYT	00617/04	00603/04							
NCL	00528/02	00437/02	00487/02	00505/02					
NCPL	00166/07	00133/07	00151/07	00153/07					
NDA	00535/05	00163/03	00262/03	00476/05	00541/05	00554/05			
NDA2	00552/05	00546/05							
NEGPM	00489/06	00351/01	00385/01	00396/06	00435/06	00440/06	00656/06		
	00004/07								
NEW1	00265/03	00270/03							
NEW2	00269/03	00258/03							
NEWS	00272/03	00224/03							
NEXT	00631/02	00636/02	00640/02	00645/02					
NEXTD	00323/06	00331/06							
NFRST	00159/05	00153/05							
NLENT	00394/04	00376/04							
NMERR	00620/01	00607/01	00641/01	00114/04	00159/04				
NODBG	00096/02	00075/02	00078/02						

NOENT 00680/02 00632/02
NOLOD 00681/03 00628/03
NPL 00526/02 00431/02 00492/02 00504/02
NSYMB 00251/02 00288/02
NTFND 00476/02 00446/02
NTMN 00724/02 00720/02
NTYPE 00529/02 00440/02 00497/02
NXFLG 00493/07 00238/02 00246/02 00265/02 00274/02 00284/02 00609/02
00649/02 00691/02 00738/02 00324/04 00359/04
NXSYM 00331/04 00448/04
NXTID 00445/02 00450/02 00455/02 00459/02 00465/02
01777 00323/07 00117/02 00577/04 00584/04 00137/07
OMITE 00313/02 00308/02
OPRND 00495/07 00368/04 00388/04 00573/04 00320/05 00373/05 00207/07
00222/07
OPWRS 00377/07 00310/06
OUTID 00708/05 00436/03 00442/03 00639/03 00567/05 00569/05 00580/05
00582/05 00607/05 00609/05 00611/05 00613/05 00615/05 00622/05
00624/05 00633/05 00639/05 00679/05 00681/05 00712/05 00727/05
OUTLN 00099/04 00091/04
OUTR 00005/02 00009/02 00014/02
PAGNO 00497/07 00568/04 00571/04 00195/07
PBE 00573/06 00559/06
PBREL 00498/07 00183/01 00521/03 00586/03 00614/03 00695/03 00596/05
PBUF 00405/07 00365/01 00393/06 00394/06 00395/06 00465/06 00468/06
00471/06 00374/07
PCNTY 00499/07 00307/01 00425/01 00452/06 00475/06
PCONT 00488/06 00454/06 00466/06 00467/06 00469/06 00470/06 00473/06
00474/06
PENTZ 00189/04 00170/04 00177/04
PGCNT 00500/07 00489/03 00514/03 00624/03 00646/03

PGMIN 00501/07 00139/01 00318/01 00517/01 00563/01 00170/02 00505/06
PIDNT 00419/07 00200/01 00541/01 00626/01 00515/02 00008/05 00030/05
00214/07
PLAD 00375/07 00430/01 00485/03 00620/03 00067/04 00069/04 00225/04
00430/04 00264/05 00269/05 00167/06 00453/06 00669/06 00705/06
00025/07
PLCNT 00502/07 00066/04 00073/04 00079/04 00085/04 00224/04 00232/04
00429/04 00438/04 00272/05 00277/05
PLEXT 00431/04 00439/04
PLFLG 00503/07 00530/01 00603/01 00646/01 00017/02 00708/03 00461/04
00561/05
PLGTH 00504/07 00118/04 00180/04 00205/04 00005/06 00144/07 00243/07
PLIST 00401/07 00666/06 00667/06 00668/06 00008/07 00009/07 00010/07
00375/07
PLMAN 00505/07 00531/03 00071/04 00074/04 00266/05
PLSCN 00226/04 00233/04
PLST 00409/07 00192/01 00269/01 00543/01 00628/01 00094/02 00370/02
00689/02 00689/04 00049/05 00199/07 00280/07
PMBUF 00490/06 00354/01 00402/06 00410/06
PPREL 00506/07 00455/03 00481/03 00500/03 00693/03 00583/05
PRBLN 00721/06 00685/06
PRCMA 00716/06 00687/06
PRENT 00507/07 00196/04 00662/04 00669/04 00621/05 00675/05 00678/05
PRENX 00249/02 00245/02
*PRHED 00465/03
PRN 00727/06 00719/06 00723/06
PROG5 00461/06 00486/06
PROG9 00485/06 00479/06
PROGL 00450/06 00379/01 00457/06 00460/06 00464/06 00481/06 00484/06
00487/06
PS1 00170/07 00164/07
PS2 00169/07 00129/07

PS3 00178/07 00125/07

PTYPE 00508/07 00339/03 00375/03 00380/03 00480/03 00528/03 00618/03
00062/04 00265/04 00265/05 00584/05 00380/06

RANAD 00510/07 00313/06 00335/06 00341/06 00343/06

RBTAD 00391/07 00192/04 00364/04 00489/04 00530/04 00607/04

RCERR 00617/01 00423/01 00474/01 00717/02 00731/02 00217/03 00229/03
00233/03 00244/03 00284/04 00310/04 00639/04 00656/04 00282/05
00565/05

RCN02 00328/01 00323/01

RCN04 00336/01 00327/01

RCN05 00342/01 00335/01

~~RCN10 00358/01 00388/01~~

RCN30 00379/01 00372/01

RCN50 00511/01 00320/01 00357/01 00361/01 00384/01

RCN60 00529/01 00632/01 00168/02 00171/02

RCNTL 00321/01 00276/01 00279/01 00349/01 00387/01 00178/02

*RCNTP 00316/01

REKEY 00512/07 00475/04 00479/04 00481/04

RELAD 00511/07 00454/03 00121/04 00173/04 00209/04 00212/04 00743/05
00014/06

REMD0 00114/06 00513/03 00558/03 00766/05 00091/06 00096/06 00103/06
00119/06

RES05 00285/06 00137/06

RES15 00139/06 00156/06 00160/06 00165/06 00176/06

RES20 00153/06 00144/06 00148/06

RES30 00166/06 00152/06 00173/06

RES40 00174/06 00171/06

RES50 00177/06 00140/06

RES60 00184/06 00204/06

RES70 00203/06 00194/06

RESET 00130/06 00515/03 00559/03 00218/06
PRIC 00513/07
RLGTH 00514/07 00499/02 00221/03
RPARN 00354/07 00415/06 00477/06
RQ 00237/05 00225/05 00229/05
RRL9 00220/06 00753/05 00208/06 00229/06
RSS 00113/04 00720/03
RTRN 00182/03 00090/03 00180/03
SAVE1 00684/03 00509/03 00596/03 00606/03
SAVE2 00685/03 00511/03 00591/03 00602/03
SAVNM 00619/02 00742/02
SAVSS 00516/07 00171/01 00670/03
SBFWA 00618/07 00160/01 00058/07
SBLWA 00619/07 00162/01 00066/07
SCANX 00297/02 00264/02
SCHK 00709/06 00704/06
SCNT 00280/03 00225/03 00277/03 00304/03 00325/03
SDSK 00634/01 00546/01 00629/01
SECT 00529/05 00293/01 00294/01 00366/03 00367/03 00172/05 00234/05
00461/05 00521/05
SECTR 00608/07 00303/01 00189/05
SELST 00361/02 00277/02 00293/02 00371/02
SET 00504/06 00631/01 00634/03 00508/06 00510/06 00517/06
SETCR 00023/06 00380/01 00393/03 00640/03 00713/05 00026/06 00062/06
00072/06 00082/06 00102/06 00115/06 00287/06
SETL 00194/07 00254/07
SETMP 00087/04 00081/04
SETOP 00369/05 00373/04 00575/04 00376/05
SETUP 00089/03 00411/01 00185/02 00100/03

SEXT 00441/04 00433/04

SIDA 00674/02 00655/02 00662/02 00693/02

SIDNT 00636/01 00542/01 00625/01

SKIP 00323/03 00456/01 00464/01 00493/01 00216/02 00313/02 00278/03
00327/03 00333/03

SLASH 00443/06 00417/06

SLSHE 00333/07 00356/01 00360/01 00383/01 00439/06

SLST 00635/01 00544/01 00627/01

SMFWA 00617/07 00156/01 00061/07

SMLWA 00603/07 00158/01 00198/01 00069/07

SPACE 00098/05 00468/03 00472/03 00504/03 00555/03 00623/03 00234/04
00263/04 00102/05

SREXT 00542/04 00548/04

SSFLG 00623/07 00168/01 00667/03 00565/07

STAT 00511/05 00502/05

STB 00627/06 00547/06 00558/06 00692/06 00695/06 00708/06 00075/07
00078/07 00079/07

STB1 00556/06 00577/06

STB2 00631/06 00654/06

STB3 00647/06 00638/06

STB4 00665/06 00728/06

STB5 00675/06 00714/06 00725/06

STBND 00545/06 00155/01 00570/06

SUBH5 00134/04 00130/04

SUBHD 00092/04 00053/04 00056/04

SV1 00270/04 00088/04 00248/04

SV2 00271/04 00090/04 00251/04

SYDBL 00497/01 00469/01

SYENT 00500/01 00467/01

SYEXT 00501/01 00471/01

SYM	00079/05	00072/05					
SYM10	00418/01	00494/01					
*SYM12	00517/07						
SYM20	00432/01	00455/01					
SYM30	00451/01	00437/01	00443/01	00450/01			
*SYM34	00518/07						
SYM40	00456/01	00460/01					
*SYM5	00519/07						
SYM50	00462/01	00427/01	00449/01				
SYM70	00464/01	00498/01	00503/01				
SYM75	00493/01	00459/01					
SYMBL	00408/01	00312/01	00381/01	00410/01			
SYNTS	00613/07	00373/07					
SYOUT	00071/05	00377/01	00550/01	00471/02	00671/02	00374/03	00629/03
	00633/03	00081/05	00090/05	00352/05	00355/05	00499/06	
SYSSC	00611/07	00239/01	00652/03				
SYSTY	00607/07	00324/01	00329/01	00333/01	00569/06	00576/06	00632/06
	00636/06						
TABS	00521/07	00741/05	00772/05	00012/06			
TADR	00522/07	00742/05	00752/05				
TBREL	00523/07	00182/01	00520/03	00525/03	00562/03	00566/03	00575/03
	00696/03	00718/03	00004/04	00011/04	00257/04	00313/05	00383/05
	00388/05	00389/05	00601/05	00614/05			
TBUF	00404/07	00252/02	00255/02	00258/02	00282/02	00340/02	00344/02
	00350/02	00351/02	00353/02	00362/02	00364/02	00366/02	00620/02
	00623/02	00627/02	00633/02	00637/02	00643/02	00680/02	00682/02
	00684/02	00307/04	00316/04	00332/04	00335/04	00338/04	00343/04
	00347/04	00351/04	00362/04	00396/04	00408/04	00412/04	00414/04
	00415/04	00419/04	00441/04	00588/05	00595/05	00599/05	00606/05
	00608/05	00612/05	00131/06	00185/06	00188/06	00191/06	
TCNT	00524/07	00259/03	00263/03	00329/03	00331/03	00756/05	00761/05
	00181/06	00203/06	00212/06	00215/06	00281/06	00293/06	00295/06
	00318/06	00330/06					
TDIR	00525/07	00311/01	00317/01	00369/01	00488/01	00093/03	00017/04

00539/05

TEMP	00526/07	00219/01	00326/01	00331/01	00345/01	00725/02	00733/02
	00160/03	00165/03	00440/03	00443/03	00459/03	00463/03	00281/04
	00301/04	00660/04	00664/04	00666/04	00112/05	00119/05	00635/05
	00640/05	00654/05	00658/05	00659/05	00687/05	00183/06	00184/06
	00186/06	00187/06	00189/06	00190/06	00192/06	00512/06	00516/06
	00634/06	00650/06					
TEMP1	00294/07	00587/01	00601/01	00556/02	00567/02	00041/03	00068/03
	00148/07	00161/07	00171/07	00178/07	00256/07		
TEMP2	00295/07	00021/04	00052/04	00494/04	00498/04	00142/07	00155/07
TERMN	00527/07	00359/01	00382/01	00242/02	00287/02	00455/06	00462/06
	00476/06	00483/06					
TESTR	00577/01	00571/01					
TIDNT	00418/07	00514/02	00715/02	00728/02	00550/03	00579/03	00300/04
	00406/04	00416/04	00019/05	00029/05	00046/05	00047/05	00053/05
	00280/05	00563/05	00370/06	00375/06			
TLOG	00512/05	00503/05					
TLST	00408/07	00268/01	00093/02	00369/02	00688/02	00677/04	00688/04
	00006/05						
TMPAD	00595/04	00534/04	00557/04	00592/04	00611/04	00614/04	
TNEXT	00528/07	00763/05	00767/05				
TPREL	00393/07	00506/03	00523/03	00568/03	00571/03	00577/03	00694/03
	00716/03	00009/04	00120/04	00134/04	00207/04	00210/04	00218/04
	00254/04	00493/04	00565/04	00668/04	00590/05	00610/05	00623/05
	00207/06	00135/07	00170/07	00173/07	00246/07	00269/07	00276/07
TR	00530/05	00218/01	00222/01	00227/01	00233/01	00287/01	00292/01
	00295/01	00299/01	00301/01	00171/05	00233/05	00460/05	00525/05
TR105	00292/05	00285/05					
TREAD	00548/01	00575/01					
TR103	00273/05	00291/05	00297/05				
TSONE	00615/07	00403/03	00720/05	00067/06	00106/06	00291/06	
TSTIN	00558/04	00539/04	00593/04				
TTBUF	00219/06	00132/06	00205/06				
TTYNO	00529/07	00074/05	00080/05				
TWORD	00530/07	00754/05	00771/05				

UBFWA 00540/07 00161/01 00181/01 00195/01 00311/05 00549/06
UBLNK 00331/07 00692/05 00320/06
UBLWA 00541/07 00163/01 00186/01 00392/05 00550/06
UDNTS 00612/07 00247/01 00662/03 00372/07
UMFWA 00542/07 00157/01 00451/03 00183/04 00560/06
UMLWA 00543/07 00159/01 00745/05 00561/06
UND 00592/04 00543/04 00552/04
UPDA1 00063/06 00073/06
UPDAT 00060/06 00054/06
UPNAM 00035/03 00309/02 00029/03 00038/03 00071/03
UPPER 00532/07 00315/05 00317/05 00242/06 00251/06 00259/06
UPPR 00066/07 00057/07
UPPR1 00071/07 00083/07
USCHK 00360/03 00349/03 00353/03
PUTCNT 00533/07
PUTRSC 00534/07
W,PRG 00506/01 00419/01 00477/01 00302/03
WAIT 00496/05 00159/05 00421/05 00480/05 00506/05 00508/05
WCMND 00176/05 00168/05
WDCNT 00536/07 00203/01 00208/01 00429/01 00454/01 00593/01 00054/03
00060/03 00230/03 00247/05 00251/05 00263/06 00269/06
WOWC 00436/05 00418/05
XPREL 00538/07 00482/03 00499/03 00401/06 00419/06 00434/06
XSSFL 00565/07 00173/01 00671/03
XXXX1 00359/04 00342/04
ZBUF 00547/07 00027/04 00031/04 00034/04 00625/04 00629/04 00633/04